End-to-End Walking Speed Estimation Method for Smartphone PDR using DualCNN-LSTM

Nobuo Kawaguchi[1], Junto Nozaki[1], Takuto Yoshida[1], Kei Hiroi[1], Takuro Yonezawa[1], and Katsuhiko Kaji[2]

[1] Nagoya University, Japan
[2] Aichi Institute of Technology, Japan

Abstract. Most of the current inertial positioning systems can be categorized as the strapdown algorithm and the step-and-heading approach. However, for the strapdown algorithm using smartphone as a sensor device, the accuracy of the current MEMS based accelerometer are not enough for estimating relative movement. Also, for the step-and-heading approach, robust estimation of step length is always difficult. In this paper, we propose an end-to-end walking speed estimation method using Deep Learning to overcome these problems. By using our method, we can achieve a smartphone PDR with higher accuracy and better robustness to gait type.

Keywords: end-to-end · inertial positioning · PDR · CNN-LSTM · DualCNN · walking speed estimation.

1 Introduction

Pedestrian Dead Reckoning(PDR) is one of the promising technology for indoor localization. Most of the current PDR techniques can be categorized as the strapdown algorithm and the step-and-heading algorithm [1]. The strapdown algorithms require precise accuracy of the sensor devices to realize the accurate localization. However, most of current smartphones equipped with MEMS sensors do not have enough precision for the double integration, it causes a time-cumulative drift-error. On the other hand, the step-and-heading PDR algorithm has the major difficulty in robust estimation of the step length and the step detection. Step length depends on several parameters such as person's height, walking speed and type of gait. So, in the conventional method [5] [15], it is not easy to estimate step length without using user dependent information. For the step detection, distinguishing "stamp" with usual "walk" is very difficult.

In this paper, we propose an end-to-end walking speed estimation method for smartphone PDR by using DualCNN-LSTM. By estimating pedestrian's walking speed directly from accelerometer sensor data, we do not have to consider about parameters such as step length, person's height nor type of gait. This means we do not have to consider about user dependent information which affects walking-speed parameters. To adapt machine learning algorithm for end-to-end speed estimation, we address two problems: 1) how we collect ground truth data of pedestrian's speed and trajectory for training data, and 2) how we design neural network for achieving high accuracy. To collect data for training, we leverage Google Tango [6] for recognizing trajectory, and we calculate pedestrian's speed data by using matrix manipulation with Karman filter. To achieve integration of step-detection and step-length estimation as speed estimation with machine learning technique, we employ LSTM(Long Short Term Memory) [14] with convolutional neural network called CNN-LSTM [12]. Additionally, we extend CNN-LSTM with the fusion of short term convolutional features and long term convolutional features. So we call our network as DualCNN-LSTM. Through our experiments, we confirmed that our method achieves higher precision such as 6.51% error rate compared to 17.55% of existing method .

2 Related Work

There are large amount of study which handles pedestrian localization systems [1]. One of the successful PDR is based on ZUPT(Zero Velocity Updates) [2] method which use fixed sensors on the foot [17]. But this method cannot utilize smartphone because it requires to fix the sensors on the foot. Most of smartphone PDR researches use step-and-heading algorithm. For the step detection, Alzantot [5] utilize finite automaton with peak detection. Also, there are several PDR competitions [15] which collects several algorithms to evaluate them under the same condition. In addition, there is a step-length estimation method which utilize stacked autoencoders [11]. These works challenged to increase accuracy of PDR. however, still not achieved enough accuracy for real-world deployment.

In addition, recent advancement of deep learning technology enables end-to-end machine learning on different domains [3] [4]. We obtain various technical hints from these researches. One of the most famous end-to-end machine learning system is "Deep Speech" [18] which enables end-to-end speech recognition. By utilizing fully connected layer and bi-directional Recurrent Neural Network, they enabled learning from unaligned transcribed audio dataset.

3 End-to-end Walking Speed Estimation for PDR

3.1 Objective

Our long-term objective is to establish a method for end-to-end PDR which inputs accelerometer and gyro sensor data and outputs relative position movement. However, this paper focuses to estimate pedestrian's speed by end-to-end manner, and calculating trajectory with the speed and heading data. Thus, we propose speed-and-heading PDR algorithm. To best of our knowledge, recognizing heading can be achieved with high accuracy. Compared with heading, current PDR methods' inaccuracy is caused by failing estimation of step length and step counts. So we divide the problem into simpler components - one is the end-to-end horizontal walking speed estimation, and the other is the horizontal heading estimation. This method is different from conventional step-and-heading approach because we don't have to estimate the step count and the step length. Our end-to-end walking speed estimation method inputs accelerometer sensor data and directly outputs terminal movement speed.

To achieve the objective, we address two problems to be solved for end-to-end speed estimation. First problem is how we prepare training data for end-to-end speed estimation. To adapt machine learning for speed estimation, we need to collect dataset which includes pedestrian's (i.e., smartphone's) trajectory, accelerometer data, gyro sensor data, and speed. It is difficult to get these data from smartphone directly, so that we leverage Google Tango and analyze it's data for preparing speed as ground truth. Second problem is how we model neural network for estimating speed. We surveyed different methods of deep neural network for activity recognition area. Through the survey and our initial experiments with different models of networks, we decided to extend CNN-LSTM for end-to-end speed estimation (see Fig.1).

3.2 PDR Data Collection for End-to-End Speed Estimation

End-to-end machine learning of PDR requires ground truth data of the precise terminal location with sensor inputs. In this paper, we employ Google Tango enabled smartphone (Lenovo PHAB2 Pro) with original location data logger software and HASC Logger [7] . Google Tango utilizes vision tracking called "VSLAM" with sensor fusion technology. By using Google Tango, we can obtain 3D trajectory of terminal position. The location measurement error of Google Tango in our pre-experiment is less than 30cm, and also in the evaluation literature[3]. So we use Google Tango tracking data as a ground truth data of the terminal location. We have collected 79 different routes by 5 subject who is equipped with 3 smartphones simultaneously. In our data collection, subjects are ordered to perform different type of gaits such as fast walk, normal walk, slow walk, and stamp. Details of the collected PDR dataset is shown in Table 1.

Based on the collected data, we have to estimate pedestrian's speed. In this paper, we focus two-dimensional trajectory. However, speed vector which is calculated from Tango's location data cannot be used directly because Tango exports data which includes 3 dimensional data. Therefore, we applied Karman filter based method [16] to estimate and remove data of gravity direction with considering noise reduction [8]. We calculate 2 dimensional moving vector by using gravity direction vector g which is estimated from Karman filter as following: $v_h = v - \frac{g \cdot v}{|g|^2} v$. We use composition of calculated speed vector as ground truth data for speed estimation in the following section. Fig. 2 shows the overview of the process of extracting horizontal speed.

3.3 DualCNN-LSTM

To model the walking speed, we employ CNN-LSTM [12] which is successfully used for activity recognition and other temporal signal processing methods. Additionally, we use fusion layer to capture short and long term features of walking

Table 1. Collected PDR Dataset

Number of Subjects	5 subjects (20's male)
Terminal Position	Hand, Left/Right Waist Pocket (3 positions)
Type of Gait	walk(fast, normal, slow, stamp), still
Total Routes	79 routes (234 files)
Average walking time	92.9 sec,,SD: 55.1sec
Average route length	52.9m, SD:35.5m

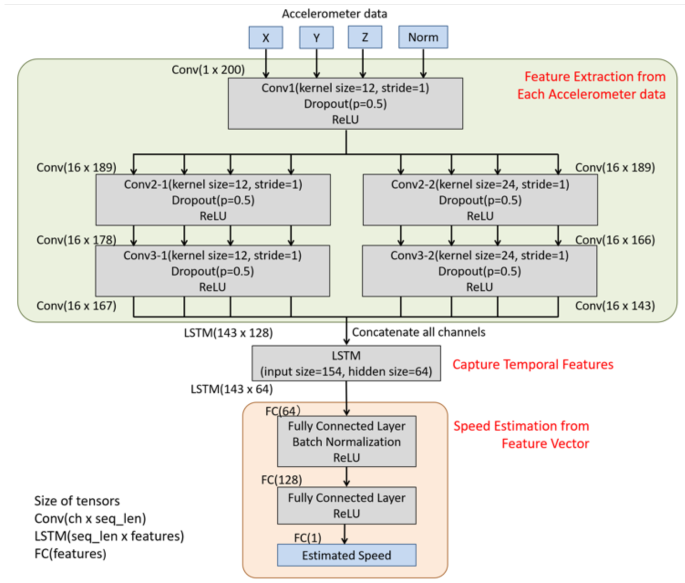

Fig. 1. DualCNN-LSTM network model for End-to-End Walking Speed Estimation

activities from the idea of local and global feature extraction [13]. Detail of the structure and tensor sizes of DualCNN-LSTM network is shown in Fig. 1. We utilize dropout(p=0.5), and ReLU for activation function.

Fig. 3 shows the data flow of the sensor data input and the estimated horizontal speed output of DualCNN-LSTM. For each 100Hz sampling timing, we input 200 samples (2.0sec) into convolutional layer of the DualCNN-LSTM network. Inside of the network, short term feature and long term feature are extracted and combined into LSTM. We use PyTorch as a deep learning platform.

4 Evaluation

We have evaluated our proposed method with conventional automaton based speed estimation method [8]. For the evaluation dataset, we use our collected PDR dataset and large indoor pedestrian sensing corpus HASC-IPSC [10]. HASC-IPSC is a corpus for indoor localization but not for real-time location estimation. So HASC-IPSC only contains 3D routes without time-stamp. Table 2 shows the detail of HASC-IPSC.

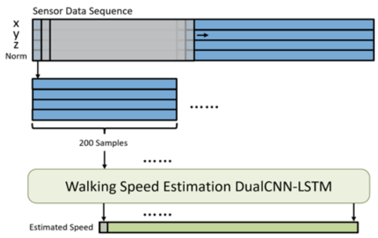

Fig. 2. Sensor data input and estimated speed output of DualCNN-LSTM

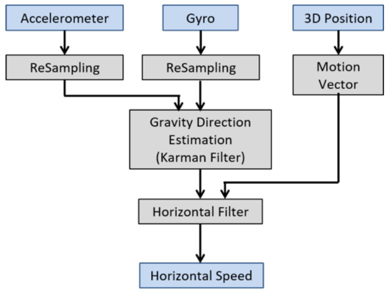

Fig. 3. Extracting horisontal speed input for the learning phase

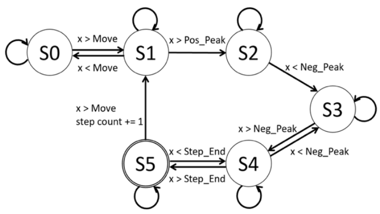

Fig. 4. Step detection finite state automaton [8]

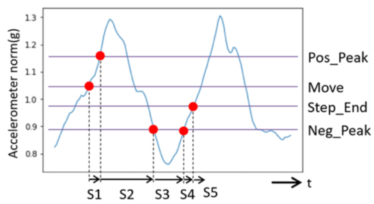

Fig. 5. State transition example of automaton based method.

4.1 Conventional Automaton based step detection [8]

For the comparison, we use conventional state-machine based PDR. Step detection finite state automaton is shown in Fig. 4. State transition of automaton by norm is depicted in Fig. 5. Parameter set for the automaton is shown in table 3.

For the input of the state machine, we use 100Hz resampled 3 axis accelerometer sensor data and norm. We suppress the high-frequency noise of the sensor data by low-pass filter using FFT (higher than 8Hz for pocket, and others for 10Hz). Additionally, we limit the least time span of steps to be more than 0.5sec to avoid error detection. For the walking speed estimation, we use step length as person's stature $\times$ 0.46.

Table 2. HASC-IPSC [10] DATASET

Subjects	100 subjects
Position	Back of waist, shirt pocket, bag
Sensors	Accelerometer, Gyro,Pressure Magetometer, WiFi
Gait	walk, still
Routes	452 (116 different routes)
Ave. time	110.1 sec,,SD: 36.0 sec

Table 3. Parameter of Step-Detection Automaton

Parameter	Hand	Pocket	HASC-IPSC
Move	1.05	1.05	1.05
Pos Peak	1.09	1.13	1.11
Neg Peak	0.97	0.89	0.93
Step End	0.98	0.96	0.97

4.2 Evaluation with PDR dataset

In this paper, we report the detailed result of our PDR dataset for evaluation. We divide the 5 subjects PDR dataset for 4 subjects for learning, and 1 subject for test, which results 198 learning files and 36 test files. For the evaluation metrics, we employ the following metrics called PIEM(Path Independent Evaluation Metrics) [9] - Average moving distance error (AMDE), Moving distance error rate for each meter (MDEM), and Moving distance error rate for each second (MDES).

For AMDE, we calculate total distance error by using estimated walking speed and elapsed time. For MDEM and MDES, we first create a scatter plot from moving distance error and ground truth distance, or elapsed time. Then we obtain the error rate from the slope of the line regressed by the least square estimate method. Result of the evaluation with our PDR dataset is shown in Table 4. All results of DualCNN-LSTM show better performance than Automaton based method. Effect of the type of gait is shown in Table 5. Automaton based method cannot handle the "stamp", so it increases estimation error. Fig. 6 and Fig. 7 show examples of the speed estimation results of conventional automaton based method and DualCNN-LSTM. Fig. 8 and Fig. 9 show the results of estimated moving paths with ground truth. To plot the moving path, we integrate estimated walking speed with moving direction which is calculated from horizontal angular velocity. Fig. 6 and Fig.8 is for "walk", and Fig. 7 and Fig.9 is for "stamp". Automaton based method cannot clearly distinguish the "stamp" with "walk", so it sometimes outputs incorrect speed in "stamp"(like in Fig. 7).

Table 4. Evaluation results with PDR dataset

Terminal Position	Metric	Proposed	Automaton based
Overall	AMDE[m]	3.83	16.78
	MDEM[%]	6.26	17.55
	MDES[%]	4.03	18.63
Hand	AMDE[m]	4.30	8.27
	MDEM[%]	6.24	15.86
	MDES[%]	4.92	8.32
L-Pocket	AMDE[m]	2.64	23.41
	MDEM[%]	4.62	20.09
	MDES[%]	2.53	26.69
R-Pocket	AMDE[m]	4.55	18.77
	MDEM[%]	7.92	16.70
	MDES[%]8	4.64	20.86

Table 5. Results with different type of gait

Type	Metric	Proposed	Automaton based
Walk	AMDE[m]	3.92	10.50
	MDEM[%]	6.10	15.92
	MDES[%]	5.66	16.53
Stamp	AMDE[m]	3.66	29.35
	MDEM[%]	82.31	518.24
	MDES[%]	2.97	19.99

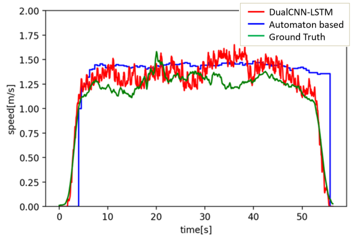

Fig. 6. Difference of walking speed estimation for "walk".

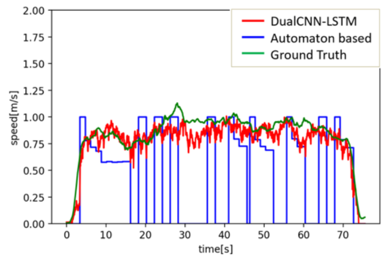

Fig. 7. Difference of walking speed estimation for "stamp".

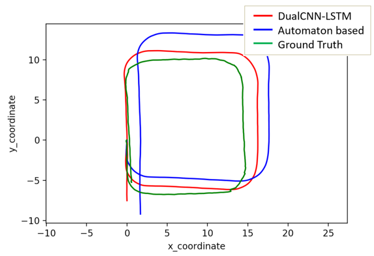

Fig. 8. Difference of estimated moving path plot of PDR "walk".

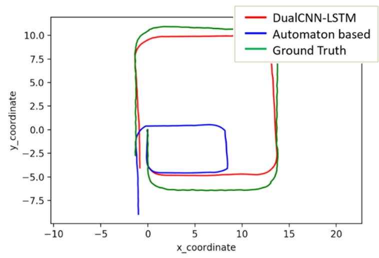

Fig. 9. Difference of estimated moving path plot of PDR "stamp".

5 Conclusion

We propose an end-to-end pedestrian walking speed estimation method using DualCNN-LSTM. By using Google Tango for collecting the corpus of pedestrian's 3D-location with sensor data, our learning method achieved higher precision such as 6.51% error rate.

Acknowledgment. This work is supported by JSPS KAKENHI Grant Number JP17H01762.

References

1. R. Harle, "A Survey of Indoor Inertial Positioning Systems for Pedestrians", IEEE Communications surveys & Tutorials, Vol.15, No.3, pp.1281-1293, 2013.
2. W. Zhang, X. Li, D. Wei, X. Ji, and H. Yuan, "A foot-mounted PDR system based on IMU/EKF+HMM+ZUPT+ZARU+HDR+compass algorithm", Int. Conf. on Indoor Positioning and Indoor Navigation(IPIN2017), 2017. doi:10.1109/IPIN.2017.8115916
3. H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-End Learning of Driving Models from Large-Scale Video Datasets, The IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp.3530-3538, 2017.

4. S. Levine, C. Finn, T. Darrell, P. Abbeel, "End-to-End Training of Deep Visuomotor Policies", Journal of Machine Learning, Vol. 16, pp.1-40, 2016.
5. M. Alzantot and M. Youssef, "UPTIME: Ubiquitous Pedestrian Tracking Using Mobile Phones", in Wireless Communications and Networking Conferene(WCNC), IEEE, pp.3204-3209, 2012.
6. R. Roberto, J. P. Lima, T. Araujo, V. Teichrieb, "Evaluation of Motion Tracking and Depth Sensing Accuracy of the Tango Tablet", Int. Symp. on Mixed and Augumented Rality(ISMAR-Adjunct), IEEE, 2016. doi: 10.1109/ISMAR-Adjunct.2016.0082
7. N. Kawaguchi, et.al, "HASC2012corpus: Large Scale Human Activity Corpus and Its Application", the Second International Workshop of Mobile Sensing: From Smartphones and Wearables to Big Data(Held with IPSN2012 and CPSWeek 2012), 2012.
8. R. Ban, K. Kaji, K. Hiroi, and N. Kawaguchi, "Indoor Positioning Method Integrating Pedestrian Dead Reckoning with Magnetic Field and WiFi Fingerprints", In 8th Int. Conf. on Mobile Computing and Ubiquitous Networking (ICMU), pp. 167-172. IEEE, 2015. doi: 10.1109/ICMU.2015.7061061
9. M. Abe, K. Kaji, K. Hiroi, and N. Kawaguchi, "PIEM: Path Independent Evaluation Metric for Relative Localization", Int. Conf. on Indoor Positioning and Indoor Navigation, pp.1-8, 2016.
10. K. Kaji, H. Watanabe, R. Ban, and N. Kawaguchi, "HASC-IPSC:Indoor Pedestrian Sensing Corpus with a Balance of Gender and Age for Indoor Positioning and Floor-plan Generation Researches", Pervasive and Ubiquitous Computing adjunct publication, pp.605-610. ACM, 2013. doi: 10.1145/2494091.2495981
11. Fuqiang Gu, Kourosh Khoshelham, Chunyang Yu, and Jianga Shang. Accurate Step Length Estimation for Pedestrian Dead Reckoning Localization Using Stacked Autoencoders. IEEE Transactions on Instrumentation and Measurement, pp. 1-9, 2018.
12. Francisco Javier Ordonez and Daniel Roggen. Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition. Sensors, Vol. 16, No. 1, 2016.
13. Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification. ACM Transactions on Graphics (TOG), Vol. 35, No. 4, pp. 110:1-110:11, 2016.
14. Sepp Hochreiter and Jurgen Schmidhuber, "Long Short-term Memory" Neural computation, Vol. 9, No. 8, pp. 1735-1780, 1997.
15. K. Kaji, K. Kanagu. K. Murao, N. Nishio, K. Urano, H. Iida, N. Kawaguchi, "Multi-algorithm on-site Evaluation System for PDR Challenge", In Proc. of Ninth International Conference on Mobile Computing and Ubiquitous Networking (ICMU2016), 2016. doi:10.1109/ICMU.2016.7742094
16. M. Kourogi, T. Ishikawa and T. Kurata, "A method of pedestrian dead reckoning using action recognition," IEEE/ION Position, Location and Navigation Symposium, Indian Wells, CA, 2010, pp. 85-89.
17. Saurabh Godha and Gerard Lachapelle. Foot Mounted Inertial System for Pedestrian Navigation. Measurement Science and Technology, Vol. 19, No. 7, pp. 1-9, 2008.
18. A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, A. Y. Ng, "Deep Speech: Scaling up end-toendspeech recognition",arXiv:1412.5567,2014.