

歩容に頑健なPDRのための 深層学習を用いた歩行速度推定手法

吉田 拓人^{1,a)} 野崎 惇登¹ 廣井 慧¹ 梶 克彦² 米澤 拓郎¹ 河口 信夫¹

概要: Inertial measurement unit (IMU) を搭載したスマートフォンの普及に伴い、慣性データから歩行者の相対位置を推定する手法である Pedestrian Dead Reckoning (PDR) の研究が盛んに行われている。しかし、多くのスマートフォンに搭載されている低コストな IMU で観測したデータにはノイズが含まれており推定結果に誤差をもたらす。さらに歩行者の歩容によって移動速度は大きく異なるため、様々な歩容に対して正しい速度を推定するアルゴリズムが必要となる。このように PDR による歩行者の相対位置推定、特に速度推定において多くの課題が存在する。そこで我々は低コストな IMU で観測したデータからでも様々な歩容に対して頑健に速度推定を行える深層学習ベースの手法を提案する。本稿では提案手法と既存の有限オートマトンベースの手法を用いて、2種類の歩容(歩行, 足踏み)に対して速度推定を行い推定結果を評価する。結果、正解速度に対する推定速度の誤差は既存手法では 17.55%、提案手法では 6.26%となり、精度の向上が確認できた。

キーワード: 深層学習, 屋内位置推定, PDR

1. はじめに

屋内位置推定は展示会における人流解析や工場における作業員の安全管理など様々な場面で必要とされている。さらに近年のスマートフォンの普及に伴い、スマートフォンの IMU を使用した PDR の研究が盛んに行われている。PDR は屋内に Wi-Fi や Beacon のようなインフラの設置をする必要のない点に特徴がある。現在研究されている多くの PDR は Strapdown algorithm と Step-and-heading algorithm に分類される [1], [2], [3]。Strapdown algorithm は正しい位置情報を推定するために高精度のデータを観測できる IMU を必要とする。しかし、現在スマートフォンに搭載されている低コストな IMU では正確な位置を推定するだけの十分な精度のデータを観測することはできず、推定位置の誤差が蓄積されてしまう。一方で Step-and-heading algorithm では歩行速度推定のためのステップ幅推定とステップ検知を行う。しかし、ステップ幅やステップの特徴は歩行者の身体的特徴や歩容の情報抜きで求めることは難しい。また歩行者のダイナミカルに変化する歩容に対して移動速度を推定するのは困難である。これらの課題を解決する最適な手法が既存の PDR においては確立されておらず、特に歩

行者の速度推定がボトルネックとなっている。以上のことから歩行者の速度推定の精度向上は PDR の精度向上にも繋がるのがわかる。

そこで我々はスマートフォンの IMU から取得した 3 軸加速度データから歩行者の移動速度を頑健に推定する深層学習ベースの手法を提案する。深層学習を用いた手法は行動認識を含む様々な分野でロバスト性向上の効果が確認されているため、PDR の速度推定にも適応できると考える。この手法は入力データが 3 軸の加速度センサのみであることが特徴であり歩行者の身体的特徴、歩容の情報は一切含まない。深層学習モデルの学習には教師あり学習を採用し正解データには歩行者の水平移動速度のノルムを用いる。正解データの収集には、スマートフォン向けの AR 技術である Google Tango[4] を利用して得られた位置情報を利用する。さらに得られた 3 次元位置情報をカルマンフィルタにより加速度と角速度から推定した重力方向ベクトルを用いて水平方向の 2 次元位置情報に変換する。また、深層学習のモデルには時系列データの特徴抽出において有用な手段として提案されている CNN-LSTM[5] を拡張して構築した。CNN-LSTM とは、Convolutional neural network (CNN) と Long Short Term Memory (LSTM) を融合させたものであり、CNN 層が時系列データの特徴を抽出し、LSTM 層が抽出した時系列データの特徴量を解析す

¹ 名古屋大学/Nagoya University

² 愛知工業大学/Aichi Institute of Technology

^{a)} takuto@ucl.nuee.nagoya-u.ac.jp

ることができる。そこで、本稿では CNN-LSTM を拡張した DualCNN-LSTM を提案する。DualCNN-LSTM はカーネルサイズが異なる 2 つの CNN を並列に分岐させ、その出力を結合する構造であり、時系列特徴量を長い系列、短い系列単位で取得できるという特徴がある。実験の結果、正解値に対する推定値の誤差率が推定速度では 17.55%、提案手法では 6.26% となり、精度の向上が確認できた。

2. 関連研究

2.1 Pedestrian localization systems

Pedestrian localization systems を扱った研究は多数存在する [1], [2], [3]。一定の成功を収めた研究の一つとして足につけたセンサで補正する ZUPT (Zero Velocity Updates) [6] をベースとした研究がある。しかし、この手法は足にセンサをつけなくてはならずスマートフォンを使用することはできない。現在、スマートフォンを用いた PDR のほとんどは Step-and-heading algorithm である。ステップ検出のために、Alzantot [7] はピーク検出を有限オートマトンで行う。ステップ幅推定では Stacked autoencoders [8] を使用した手法がある。このように PDR の精度向上のため、様々な手法が提案されている。しかし、どの研究も実用化に十分な精度はまだ得られていない。

2.2 End-to-end Machine Learning System

近年の深層学習の技術は様々なドメインにおいて End-to-end の機械学習を可能している [9], [10], [11]。我々はこれらの研究から様々なヒントを得た。例えば、Speech Recognition を End-to-end で可能にした Deep Speech [12] である。Deep Speech は全結合層と bi-directional Recurrent Neural Network を結合したモデルを用いて音声データセットから学習を可能にした。また、行動分野においても深層学習を利用した手法が増加している。Ordóñez らの研究 [13] では、複数の畳み込み層とリカレント層を用いた構造の深層学習モデルを作成し行動認識を行なっている。この研究では、センサデータを短いセグメントに分割しモデルに入力し、セグメント毎に認識を行うことを繰り返して予測を行う。Peng らの研究 [14] では、複数の畳み込み層を用いた構造の深層学習モデルを用いて歩く、止まるなどの単純な行動の認識を行い、畳み込み層の出力をリカレント層に入力し買い物といった複雑な行動の認識を行う。これらの畳み込み層とリカレント層による行動認識モデルと時系列データを一定時間のセグメントごとの扱う手法を応用して、時系列の 3 軸加速度データから歩行者の移動速度を推定する研究にも応用できると考えた。

3. 提案手法

PDR によって歩行者の位置推定を行う際、大きく分けて移動速度、進行方向推定の 2 つから成る。既存の PDR

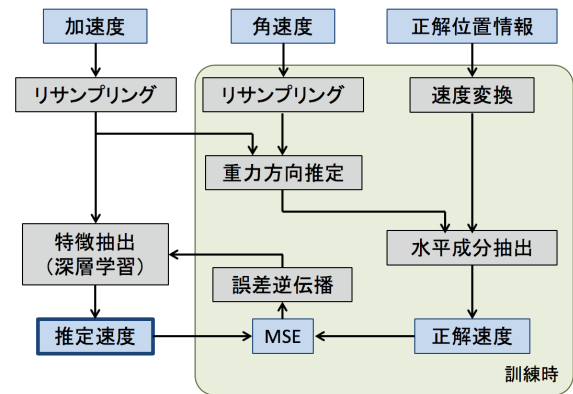


図 1: 速度推定の処理フローチャート

において、進行方向推定の多くは角速度の時間積分によって行なっている。精度向上のため角速度データのノイズ除去 [15] やマップマッチングなどの研究もなされており、一定の精度は出している。一方の移動速度推定においては、既存の PDR ではステップ検出を誤ると大きく位置情報を誤る可能性がある。そのため、既存の PDR においてボトルネックとなりやすいのは移動速度推定である。そこで、本稿では移動速度推定の精度、ロバスト性向上を目的とする。

我々は移動速度推定の精度、ロバスト性向上のために深層学習ベースの手法を有効であると考え、深層学習を用いた End-to-end の移動速度推定手法を提案する。提案手法の全体像を図 1 に示す。ネットワークの学習には加速度、角速度、正解位置情報を用いる。推定時には、加速度センサデータのみから移動速度を推定する。モデルを学習して、速度を推定する前に行う行程は以下の 2 つである。

- データ収集と正解データの生成
- モデルの構築

入力データである加速度、角速度センサデータは一般的な Android 端末を使用した。しかし正解位置情報を一般のスマートフォンで取得することは難しい。そこで我々は正解位置情報取得のため Google Tango [4] を使用し、そのデータから水平速度成分を取得しそれを正解データとすることにした。モデルの構築には CNN-LSTM [5] を拡張した DualCNN-LSTM を提案する。

3.1 データ収集と正解データの生成

移動速度の推定を行うモデルの学習のためには正確な端末の位置情報と入力データとなる加速度、角速度が必要となる。そこで本稿では正解位置情報を Google Tango (Lenovo PHAB2 Pro)、入力データは HASC Logger (Xperia G8342, Nexus 6) を用いて取得する (ただし、保持位置が手のデータに関しては Google Tango で正解データと入力データの両方を計測した)。Google Tango は加速度や角速度に加えてカメラや深度センサを用いることで、端末の向きや移動量を高精度に推定でき、計測開始時を原点として相対位置

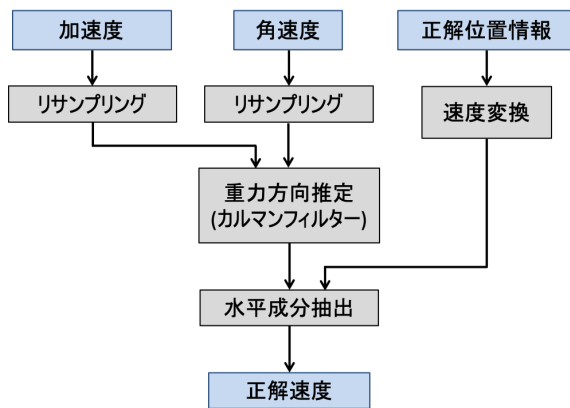


図 2: 正解速度算出のフローチャート

表 1: データセットの概要

被験者数	5名 (20代男性)
端末保持位置	手持ち, パンツの左右ポケット
行動種別	歩行, 足踏み
延べ経路数	79
各経路の歩行時間	Avg: 92.89 秒, SD: 55.05 秒
各経路の長さ	52.93 m, SD:35.53 m
端末	PHAB2 Pro (Android 6.0.1) Xperia G8342(Android 8.0.0) Nexus 6 (Android 6.0.1)
ソフトウェア	Google Tango HASC Logger for Android
センサデータ	加速度 角速度 GPS 磁気 気圧 WiFi

を取得できる。しかし、3次元位置情報のままではモデルの学習に使えないので水平速度を抽出する必要がある。そこで、正解位置情報と同時に取得した加速度と角速度データからカルマンフィルタを用いて重力方向を推定する。そして求めた重力方向 g と3次元移動速度 v を用いて以下の式より移動速度の水平成分 v_h を算出する:

$$v_h = v - \frac{g \cdot v}{|g|^2} v. \quad (1)$$

算出された水平速度ベクトルのノルムを正解速度として利用する。流れを2に示す。我々は5名の被験者から同一の経路を時計回り、反時計回りの様々な歩容で計測し、合計79の経路を取得した。スマートフォンの保持位置は手、左右のポケットの計3カ所、歩容は歩行、足踏みの2種類。データの詳細は表1に示す。

3.2 DualCNN-LSTM

深層学習のモデルには時系列データの特徴抽出において有用な手段として提案されている CNN-LSTM[5] を拡

張して構築した。CNN-LSTM とは、Convolutional neural network (CNN) と Long Short Term Memory (LSTM) を融合させたものであり、CNN 層が時系列データの特徴を抽出し、LSTM 層が抽出した時系列データの特徴量を解析することができる。そこで、本稿では CNN-LSTM を拡張した DualCNN-LSTM を提案する。DualCNN-LSTM はカーネルサイズが異なる2つの CNN を並列に岐分させ、その出力を結合する構造であり、時系列特徴量を長い系列、短い系列単位で取得できるという特徴がある。歩容や歩行者によって観測する慣性データの周期が異なる、そこで異なる時間幅で特徴量を抽出することが精度向上に繋がると考えたため、この岐分かれ構造を採用した。サイズを図3に示した。1層目の畳み込み層で加速度センサのデータ特徴量を抽出して、2層目以降で岐分かれの畳み込み層によって2種類の特徴量を抽出する。畳み込み層 2-1 と 3-1 のカーネルサイズは 12、畳み込み層 2-2 と 3-3 のカーネルサイズは 12 の倍である 24 となっている。つまり、前者の畳み込み層では短い時系列データを、後者の畳み込み層では長い時系列データの特徴を抽出することが可能である。さらにこれら2種類の特徴量を結合して、LSTM 層入力する。LSTM 層では畳み込み層で抽出した時系列特徴データを学習し、特徴ベクトルを出力する。その特徴ベクトルを用いて全結合層が速度を出力する。図4には3軸加速度とそのノルムを DualCNN-LSTM に入力して出力を得るまでのフローチャートが示してある。センサデータはすべて 100Hz でリサンプリングされており、そのデータの 200 サンプル (2 秒分) を1つの入力データとした。入力データに対応する正解データは、その2秒の間に観測した正解位置情報から算出した水平速度の平均である。

4. 評価

4.1 比較対象とする移動推定手法

提案手法の比較対象として、既存の有限オートマトンによるステップ検知を用いた速度推定手法 [7] を用いる。入力するセンサデータは、提案手法と条件を揃えるため、100Hz にリサンプリングした加速度、角速度信号を用いる。これに加え、FFT を用いたローパスフィルタにより一定以上の周波数成分をカットする処理を行う。カットオフ周波数は今回収集したデータセットの端末保持位置ごとに設定し、手持ちでは 10Hz、パンツの左右のポケットでは 8Hz とした。ポケットに入れた端末では手持ちに比べ細かい振動が多く含まれるため、カットオフ周波数を低く設定した。次に、ローパスフィルタをかけた加速度信号を用いてステップ検知を行う。ステップ検知には図5に示す有限オートマトンを利用する。図中の x は加速度ノルムの値であり、状態遷移の条件となる各閾値はそれぞれ Move:動作しているか否か、Pos Peek: 極大、Neg Peek: 極小、Step End: 歩行の終了を意味する。各閾値は、手持ち、パンツの左右のポケッ

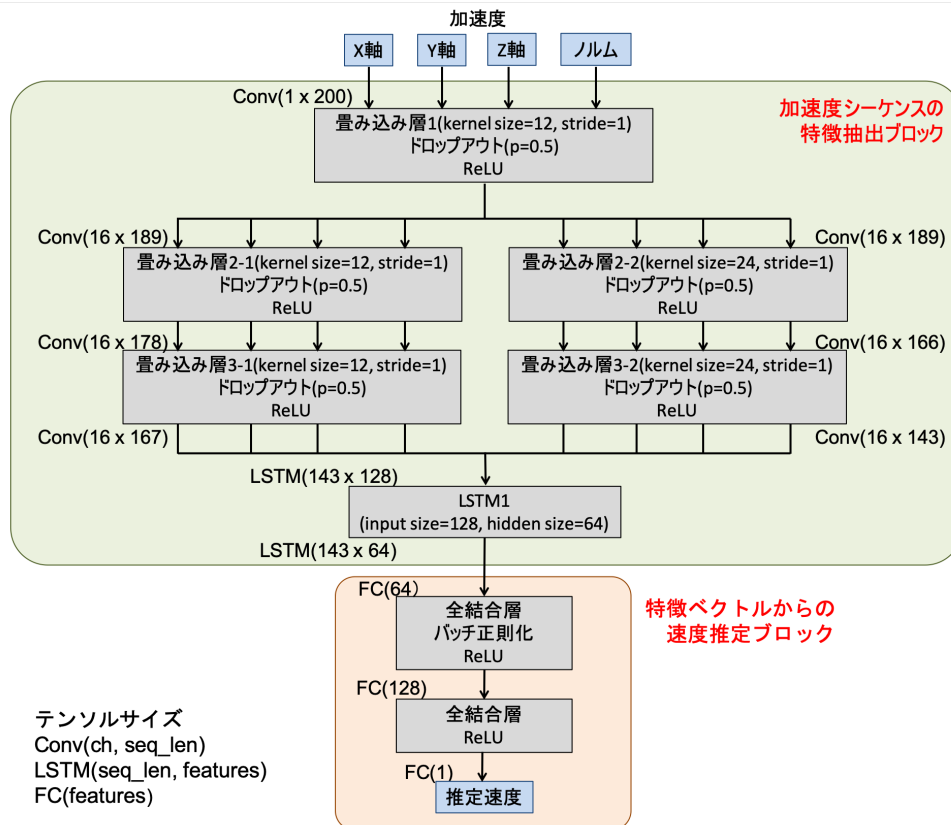


図 3: DualCNN-LSTM による歩行速度推定モデル

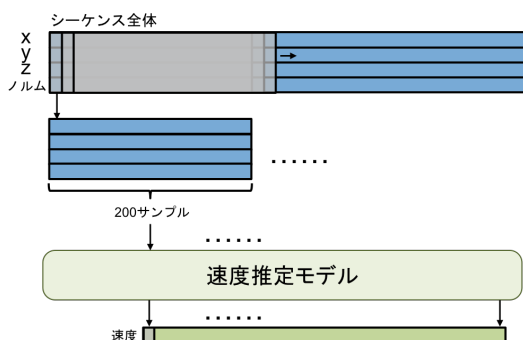


図 4: 2 秒間のセグメントごとの入力模式図

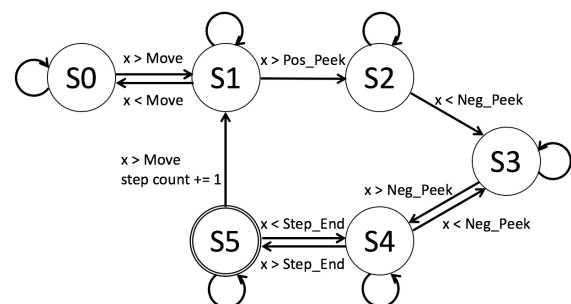


図 5: ステップ検知の有限オートマトン

トの各端末保持位置ごとに調整し、表 2 に示す通りに設定した。最後に、検出されたステップから移動速度を推定し位置を算出する。一步あたりの歩幅は身長× 0.46m で推定し、直前のステップからの経過時間で割り移動速度を算出する。算出された移動速度は最大サンプル数 4 のキューに入れ、その時刻におけるキュー内の平均を移動速度として出力する。

4.2 検証データセットによる評価結果

4.2.1 評価指標

モデルの評価のためにデータセットを訓練用に 4 人、テスト用に 1 人となるように分ける。評価指標には Path Independent Evaluation Metrix (PIEM)[16] を使用した。

表 2: オートマトンの各閾値

	手	ポケット (左右)
Move	1.05	1.05
Pos Peek	1.09	1.13
Neg Peek	0.97	0.89
Step End	0.98	0.96

PIEM では以下の 3 つの指標で評価を行う。

- (1) 総距離誤差の平均:
Average moving distance error (AMDE)
- (2) 正解移動距離あたりの推定移動距離の誤差率:
Moving distance error rate for each meter (MDEM)
- (3) 歩行時間あたりの推定移動距離の誤差率:
Moving distance error rate for each second (MDES)

AMDE は推定された歩行速度と経過時間を使用して総距離誤差を計算する。MDEM と MDES では、最初に移動距離誤差と正解距離、または移動距離誤差と経過時間から散布図を作成する。次に最小二乗推定法で回帰した直線の傾きから誤り率を求める。これら3つの評価指標は値が小さいほど推定結果の精度が高い事を示す。

4.2.2 評価結果

結果を表3に示す。全体の評価結果を見ると、提案手法によって MDEM が 17.55% から 6.26% に改善、MDES が 18.63% から 4.03% に改善したことがわかる。さらに、手持ち、左右のポケットそれぞれの項目ごとに見てもすべての評価指標において精度が改善している。表4に行動種別ごとの評価結果を示す。歩行、足踏みの項目両方において提案手法の方が既存手法より精度が高いことが確認できる。歩行では MDEM が 15.92% から 6.10% に改善、MDES が 16.53% から 5.66% に改善、足踏みでは MDEM が 518.24% から 82.31% に改善、MDES が 19.99% から 2.97% に改善している。足踏みにおいて提案手法、既存手法の MDEM の値が極めて大きいのは足踏みでの移動距離はほぼゼロに等しいため、わずかな推定誤差で正解距離あたりの誤差は極めて大きくなるのが原因である。また、足踏み時の AMDE、MDES の値を既存手法と提案手法で比較すると、既存手法では歩行時に比べて精度を落としているのに対して、提案手法においては歩行時に比べて精度の低下が見られない。このことから提案手法が歩容に対して頑健であることが確認できる。

次に歩行データにおいて提案手法と既存手法の間で異なる推定結果を示す例を2つ挙げる。一つは図6の提案手法の方が既存手法より速度の立ち上がり、立ち下がり、正確にトラッキングできている例、もう一つは図7の既存手法がステップの検知を失敗している例である。図6aと図7aに、提案手法と既存手法の速度推定結果の例を示す。図6bと図7bは、提案手法、既存手法によって推定されて速度と正解速度による移動軌跡の推定結果を示す。移動経路をプロットするために、推定速度と正解速度に加えて水平方向の角速度から計算される進行方向ベクトルを用いる。

図6aでは提案手法と既存手法が共に正解速度の概形がトラッキングできている。しかし、提案手法の方が時刻5秒周辺の速度の立ち上がり、時刻45秒周辺の立ち下がり、既存手法よりも正確にトラッキングできている。この違いは図6bからも確認できる。既存手法もおおよそその経路の外形は捉えられているが速度変化が大きい部分の推定を行っていないため、最終的な推定位置が正解位置に対して提案手法よりも離れている。

図7aでは既存手法がステップ検知を誤っている。既存手法はステップ検知を行えないと移動速度を 0 m/s と出力するため、図7aのように推定移動経路を大きく誤ってしまい、推定経路も図7bに示す通り大きく誤ってしまう。ス

表3: 検証データセットに対する評価結果 (FA: Finite Automaton)

端末保持位置	評価値	DualCNN-LSTM	FA
全体	AMDE[m]	3.83	16.78
	MDEM[%]	6.26	17.55
	MDES[%]	4.03	18.63
手持ち	AMDE[m]	4.30	8.27
	MDEM[%]	6.24	15.86
	MDES[%]	4.92	8.32
左ポケット	AMDE[m]	2.64	23.41
	MDEM[%]	4.62	20.09
	MDES[%]	2.53	26.69
右ポケット	AMDE[m]	4.55	18.77
	MDEM[%]	7.92	16.70
	MDES[%]	4.64	20.86

表4: 行動種別ごとの評価結果 (FA: Finite Automaton)

歩容	評価値	DualCNN-LSTM	FA
歩行	AMDE[m]	3.92	10.50
	MDEM[%]	6.10	15.92
	MDES[%]	5.66	16.53
足踏み	AMDE[m]	3.66	29.35
	MDEM[%]	82.31	518.24
	MDES[%]	2.97	19.99

テップ誤検知の原因として、オートマトンベースの既存手法ではあらかじめ設定した閾値によって、ステップ検知がうまく行える歩行速度や歩容が存在するからであると考えられる。この点からも提案手法の方が歩容に頑強であることが確認できる。

5. おわりに

本稿では、歩容に頑強かつ高精度な PDR を実現するため、深層学習を用いた End-to-end の歩行者移動速度推定手法を提案した。我々は時系列データの特徴抽出のために DualCNN-LSTM の提案し、モデルの学習のために Google Tango を用いて慣性データと正解位置情報を含むデータセットの作成を行なった。実験の結果、評価指標 MDEM においては誤差率を 17.55% から 6.26% へ改善、MDES においては誤差率を 18.63% から 4.03% へ改善を達成した。さらに、歩容ごとの評価結果から提案手法が既存手法よりも歩容に対して頑健に速度推定を行えることがわかった。今後はデータセットの歩容のバリエーションを増やし、さらなる検証を行う。また、方向推定を深層学習で行うためのアプローチも検討する必要がある。将来的には一つのモデルで速度、方向を推定を行えることが理想である。

謝辞 本研究は JSPS 科研費 (JP17H01762) の助成を受けたものである。

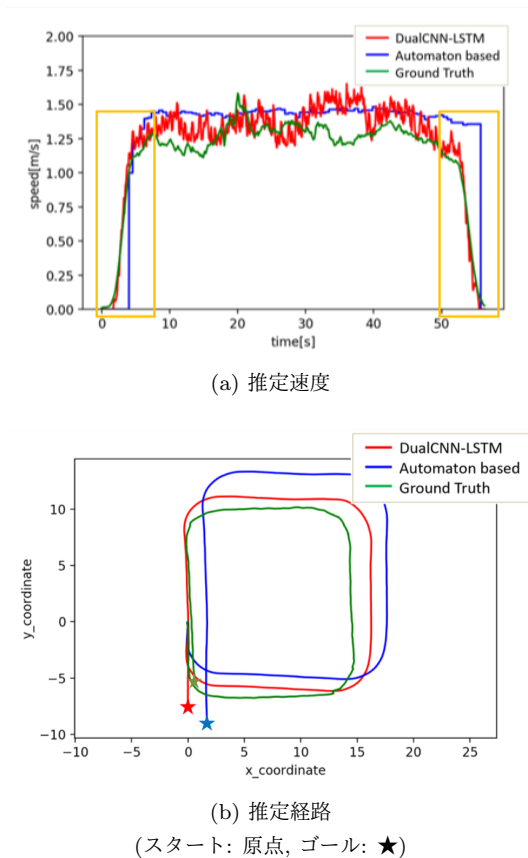


図 6: 速度変化に対する推定精度の比較

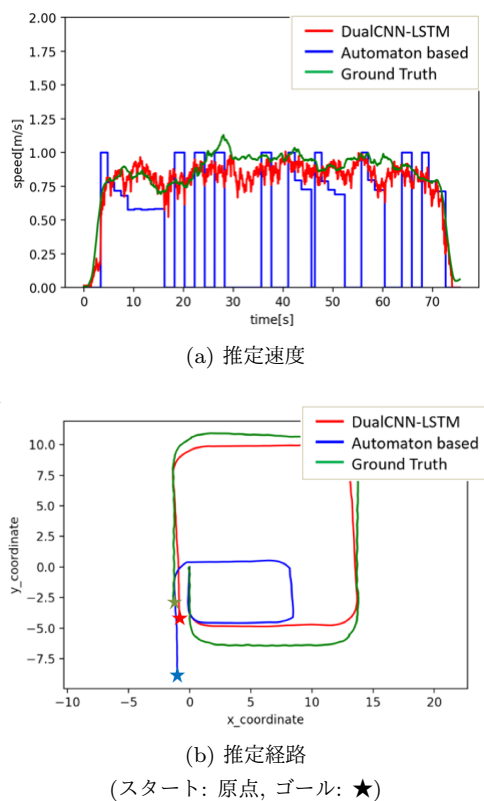


図 7: 既存手法がステップの検知を失敗している例

参考文献

[1] Harle, R.: A Survey of Indoor Inertial Positioning Sys-

tems for Pedestrians, *IEEE Communications Surveys Tutorials*, Vol. 15, No. 3, pp. 1281–1293 (2013).

[2] Correa Vila, A., Barceló, M., Morell, A. and Lopez Vicario, J.: A Review of Pedestrian Indoor Positioning Systems for Mass Market Applications, *Sensors (Switzerland)*, Vol. 17 (2017).

[3] Sakpere, W., Oshin, M. A. and Mlitwa, N.: A State-of-the-Art Survey of Indoor Positioning and Navigation Systems and Technologies, *South African Computer Journal*, Vol. 29, No. 3 (online), available from <http://sacj.cs.uct.ac.za/index.php/sacj/article/view/452> (2017).

[4] Roberto, R., Lima, J. P., Araújo, T. and Teichrieb, V.: Evaluation of Motion Tracking and Depth Sensing Accuracy of the Tango Tablet, *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 231–234 (2016).

[5] Francisco Javier Ordóñez, Roggen, D.: Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition, *Sensors*, Vol. 16, No. 1, p. 115 (2016).

[6] Zhang, W., Li, X., Wei, D., Ji, X. and Yuan, H.: A foot-mounted PDR system based on IMU/EKF+HMM+ZUPT+ZARU+HDR+compass algorithm, *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–5 (2017).

[7] Alzantot, M. and Youssef, M.: UPTIME: Ubiquitous pedestrian tracking using mobile phones, *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 3204–3209 (2012).

[8] Gu, F., Khoshelham, K., Yu, C. and Shang, J.: Accurate Step Length Estimation for Pedestrian Dead Reckoning Localization Using Stacked Autoencoders, *IEEE Transactions on Instrumentation and Measurement*, pp. 1–9 (2018).

[9] Xu, H., Gao, Y., Yu, F. and Darrell, T.: End-to-End Learning of Driving Models from Large-Scale Video Datasets, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3530–3538 (2017).

[10] Kai Wang, Babenko, B. and Belongie, S.: End-to-end scene text recognition, *2011 International Conference on Computer Vision*, pp. 1457–1464 (2011).

[11] Levine, S., Finn, C., Darrell, T. and Abbeel, P.: End-to-end Training of Deep Visuomotor Policies, *J. Mach. Learn. Res.*, Vol. 17, No. 1, pp. 1334–1373 (online), available from <http://dl.acm.org/citation.cfm?id=2946645.2946684> (2016).

[12] Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A. and Y. Ng, A.: DeepSpeech: Scaling up end-to-end speech recognition (2014).

[13] Ordóñez, F. J. and Roggen, D.: Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition, *Sensors*, Vol. 16, No. 1 (online), available from <http://www.mdpi.com/1424-8220/16/1/115> (2016).

[14] Peng, L., Chen, L., Ye, Z. and Zhang, Y.: AROMA: A Deep Multi-Task Learning based Simple and Complex Human Activity Recognition Method Using Wearable Sensors, *ACM Interactive, Mobile, Wearable and Ubiquitous Technologies*, Vol. 2, No. 2, pp. 74:1–74:16 (online), DOI: 10.1145/3214277 (2018).

- [15] Manos, A., Klein, I. and Hazan, T.: Gravity-Based Methods for Heading Computation in Pedestrian Dead Reckoning, *Sensors*, Vol. 19, p. 1170 (2019).
- [16] Abe, M., Kaji, K., Hiroi, K. and Kawaguchi, N.: PIEM: Path Independent Evaluation Metric for Relative Localization, *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8 (2016).