OCEAN: A new opportunistic computing model for wearable activity recognition

Yiqiang Chen[1,2], Yang Gu[1,2,3], Xinlong Jiang[1,2,3], Jindong Wang[1,2,3]

[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China

[2]Beijing Key Laboratory of Mobile Computing and Pervasive Device, Beijing, China

[3]University of Chinese Academy of Sciences, Beijing, China

{yqchen, guyang, jiangxinlong, wangjindong }@ict.ac.cn

UbiComp/ISWC'16 Adjunct, September 12-16, 2016, Heidelberg, Germany
ACM 978-1-4503-4462-3/16/09.
http://dx.doi.org/10.1145/2968219.2971453.

Abstract

Activities of Daily Living (ADL) recognition through wearable devices is an emerging research field. While, for many applications, recognition methods are faced with simultaneously dynamic changes in feature dimension, activity class and data distribution. Existing approaches mainly handle at most one of these three challenges, which significantly affects their performance. In this paper, we propose an Opportunistic Computing model for wEarable Activity recognitioN (OCEAN); by fusing random mapping, fuzzy clustering, and weight updating techniques, OCEAN can online adaptively adjust Single-hidden Layer Feedforward neural network's connection, structure and weight in a coherent manner. Experimental evaluations demonstrate that OCEAN improves the recognition accuracy by 5% to 15% compared to traditional approaches towards dynamic changes.

Author Keywords

Activity recognition; online learning; opportunistic computing; neural network.

ACM Classification Keywords

H.5.m. Information Systems: Information interfaces and presentation: Miscellaneous.

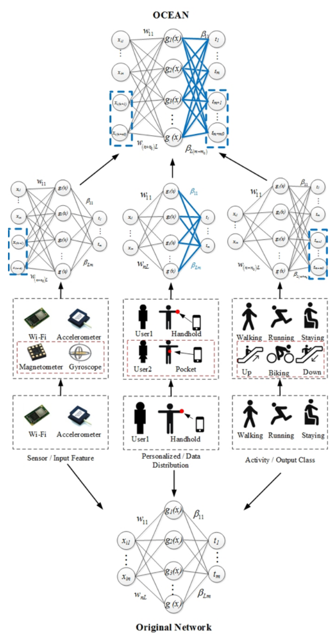

Figure 1. Framework of OCEAN

Introduction

The rapid development of computing chip and hardware has facilitated the proliferation of sensors and wearable devices. And the ubiquity of smart wearable devices has changed people's life in many aspects, especially in Activities of Daily Living (ADL) recognition.

Currently, mainstream approaches for ADL recognition are adopting fixed model (batch learning) [1] to recognize pre-defined activities [2]. However, when dynamic changes occur in real situations, such as opportunistic sensors, undefined activities, and different users, performance of fixed model deteriorates. Though, existing online learning methods can utilize existing model and newly arrived data to efficiently generate a new model, it can only handle certain change, cannot learn or be self-adaptive when feature dimension, activity class and user style change simultaneously [3-5], which always occurs in real situations. Hence, in this paper, we propose a new online learning method, Opportunistic Computing model for wEarable Activity recognitioN (OCEAN), which integrates random mapping, fuzzy clustering, and weight updating to form a dynamic learning neural network, so as to accomplish high recognition accuracy towards dynamic changes simultaneously.

Methodology

The framework of proposed OCEAN is illustrated in Figure 1. To train an initial model, there are N_0 arbitrary distinct samples $(x_i, t_i) \in \Re^n \times \Re^m$, $\mathrm{i} = 1,2,\dots,N_0$. Here, x_i is a $n\times1$ input vector $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T$ and t_i is a $m\times1$ target vector $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T$. When N_1 incremental data $(x_i', t_i') \in \Re^{n'} \times \Re^{m'}$ arrive to update the model, where $x_i' = [x_{i1}', x_{i2}', \dots, x_{in'}']^T$, $t_i' = [t_{i1}', t_{i2}', \dots, t_{im'}']^T$. If $n' > n$, which means feature dimension is changed, then ***random mapping*** technique is adopted; similarly; if $m' > m$, which indicates activity class is changed, then ***fuzzy clustering*** techniques is adopted; and when different user styles lead to the change of data distribution, ***weight updating*** technique is adopted.

Random mapping

Existing network cannot work well when feature dimension changes. However, we can handle this issue by random mapping technique according to [3], which randomly generates connection between input layer and hidden layer to deal with various feature dimension. Assuming $n' > n$, which implies feature dimension is added, then we bring in an input weight transfer matrix *P*, and a weight supplement vector Q_i to adapt to the new connection, which is $\{a_i' = a_i \cdot P + Q_i\}_{i=1}^{L}$, where a_i is the weight connection between input nodes and hidden nodes, a_i' is the new weight connection. For matrix *P*, if $P_{ij} = 1$, it means after the changing, the i^{th} dimension of original feature becomes the j^{th} dimension of new feature. Q_i supplements the corresponding input weight for the new adding features. To make it clear, if the feature of original training data is $\{\mathrm{F}_1, \mathrm{F}_2\}$, and feature of incremental data is $\{\mathrm{F}_1, \mathrm{F}_2, \mathrm{F}_3, \mathrm{F}_4\}$, then we can generate matrix *P* and Q_i to solve the changing feature dimension problem, where $P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$, $\mathrm{Q_i} = \{0, 0, \mathrm{Q}_3, \mathrm{Q}_4\}$; and $\mathrm{Q}_3, \mathrm{Q}_4$ can be generated randomly without time-consuming fine tuning.

Fuzzy clustering

With time passing by, user may have some new activities which are not included in the existing model. If we adopt old model to classify new activities, which would be wrongly classified to one of existing classes.

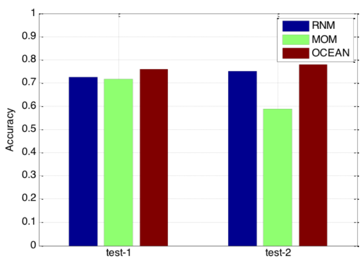

Figure 2. Results for changing feature dimension

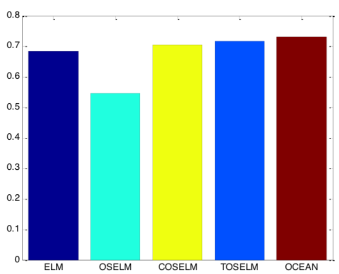

Figure 3. Results for changing data distribution

In order to maintain the performance of recognition model, we put forward fuzzy clustering technique, which can discover new activity classes through probability based clustering with unlabeled data, and then update the model with new activity classes. Fuzzy clustering technique has four steps [4]: 1) Feature extraction—extracting features from incremental data (mostly unlabeled) of different sensors. 2) Outlier detection—adopting one-class classification algorithm to isolate and accumulate initially unknown but increasingly distinguishable data; 3) Fuzzy clustering—discovering new cluster candidates using fuzzy clustering algorithm; 4) Class-incremental update—updating existing model with increasing classes.

Weight updating

Different users have personalized wearable styles and activity styles, which leads to inconsistent data distribution for the same activity. Traditional learning methods assume that data distribution maintains the same; when data distribution changes, performance decreases. Here, we use weight updating technique to solve data distribution problem. According to [5], the relationship for new model, existing model and incremental data satisfies: $\beta^{(new)} = \beta^{(old)} + \Delta\beta(x^*)$. When the distribution of incremental data and existing data are different, a suitable penalty ω to $\Delta\beta(x^*)$ is brought in, so that new model can balance the contribution between incremental data and existing model to improve the performance. The penalty ω is adaptively calculated by the central tendency and dispersion characteristics of incremental data [5].

Experiments

In order to test the performance of the proposed OCEAN, we select a behavior recognition dataset from HASC Challenge 2012 [6] which has 31844 data. The dataset uses 3-axis accelerometer and 3-axisgyroscope to identify six different behaviors: stay (Label 1), walk (Label 2), jog (Label 3), upstairs (Label 4), downstairs (Label 5), and skip (Label 6).

Firstly, we validate the performance when feature dimension changes. Here are two situations: for test-1, 14-dimension features of training data are extracted from 3-axis accelerometer, while incremental data and testing data add another 14-dimension features extracted from 3-axis gyroscope; for test-2, 14-dimension features of training data are extracted from 3-axis gyroscope, while incremental data and testing data add another 14-dimension features extracted from 3-axis accelerometer. Three methods are compared in Figure 2: 1) RNM means retraining a new model; 2) MOM means maintaining the old model; and 3) OCEAN indicates updating the old model to fit the new feature dimension with incremental data. Results show that for both test-1 and test-2, OCEAN has the highest accuracy. Compared with other methods, OCEAN not only maintains the knowledge of old model, but also takes use of new incremental features to improve accuracy.

Then, we validate the performance when activity class changes. Four activities are chosen as known classes and the rest two activities are unknown classes. We mainly compare the performance of OCEAN with ELM [7] for its fast speed and without fine tuning which is suitable for wearable computing. Firstly, batch learning ELM and online learning OCEAN have the same initial recognition accuracy (83.00% on average). When unknown classes appear, accuracy of batch learning degenerates greatly. Specifically, the average accuracy

	1	2	3	4	5	6
1	**97.29**	0.58	0.58	0.39	0.77	0.39
2	0.69	**62.84**	3.26	21.55	9.34	2.32
3	0	0	**93.03**	0.54	0.98	5.45
4	1.48	1.48	5.30	**72.67**	8.90	10.17
5	0.64	5.21	5.63	21.27	**61.83**	5.42
6	0	0	3.47	5.90	0.63	**90.00**

Table 1. Result (77.33%) for OCEAN when changes occur simultaneously

	1	2	3	4	5	6
1	**94.70**	1.02	0.93	1.21	1.58	0.56
2	--	--	--	--	--	--
3	0.17	2.15	**79.88**	1.24	11.34	5.22
4	0.38	34.93	1.08	**32.99**	28.68	1.94
5	--	--	--	--	--	--
6	0.08	3.81	3.56	5.97	8.29	**78.29**

Table 2. Result (60.87%) for batch learning ELM when changes occur simultaneously

of batch-mode ELM is 66.90% for one new class, and accuracy declines to 55.66% (ELM) for two new classes. While the corresponding accuracy for OCEAN is 71.00% and 76.2% respectively.

Then, we validate the performance when data distribution changes. Here, we mainly compare OCEAN with batch learning ELM and online learning methods OSELM [8], COSELM [9], TOSELM [5]. The recognition performance is shown in Figure 3. In Figure 3, ELM and OSELM cannot handle the changing of data. And, the other three methods have similar better accuracy than ELM and OSELM.

Finally, when feature dimension, activity class and data distribution change simultaneously, the result of proposed OCEAN is given in Table 1. We first utilize random mapping to solve the feature dimension changing from 14 to 28, then use fuzzy clustering to discover two new activity classes (Label 2 and Label 5), finally adopt weight updating technique to handle the inconformity of data distribution. Results show that OCEAN not only maintains the knowledge from the old model, but also learns new knowledge from incremental data towards changes, it can achieve 77.33% accuracy on average. In comparison, even if we utilize all the 28-dimension features for batch leaning ELM (Table 2), as it cannot deal with the changing activity class or data distribution, its accuracy is 60.87% on average.

Conclusion

In this work, we propose an online leaning scheme for activity recognition which works well under dynamic changes in feature dimension, activity class and data distribution simultaneously. Compared with traditional online learning method, it is more flexible. Experiments based on open dataset demonstrate the effectiveness and robustness of proposed method.

Acknowledgement

This work was supported by Natural Science Foundation of China (No.61572471) and National Key Research and Development Plan (No.2016YFB1001200).

References

1. Thomaz E, et al. A practical approach for recognizing eating moments with wrist-mounted inertial sensing. UbiComp, 2015: 1029-1040.
2. Khan A, et al. Beyond activity recognition: skill assessment from accelerometer data. UbiComp, 2015: 1155-1166.
3. Jiang X, et al. Feature adaptive online sequential extreme learning machine for lifelong indoor localization. Neural Computing and Applications, 2016, 27(1): 215-225.
4. Chen Z, et al. ContextSense: unobtrusive discovery of incremental social context using dynamic bluetooth data. UbiComp, 2014: 23-26.
5. Gu Y, et al. TOSELM: timeliness online sequential extreme learning machine. Neurocomputing, 2014, 128: 119-127.
6. Kawaguchi N, et al. Hasc2012corpus: Large scale human activity corpus and its application. IPSN, 2012, 12.
7. Huang G, et al. Extreme Learning Machine: Theory and Applications. Neurocomputing,2006: 489–501.
8. Liang N Y, et al. A fast and accurate online sequential learning algorithm for feedforward networks. Neural Networks, 2006, 17(6): 1411-1423.
9. Gu Y, et al. Constraint online sequential extreme learning machine for lifelong indoor localization system. IJCNN), 2014: 732-738.