Patient's Motion Recognition Based on SOM-Decision Tree

Wei Yu, Hongli Yan, Junqi Guo*, and Rongfang Bie

College of Information Science and Technology, Beijing Normal University, Beijing, China
weiyu@cupes.edu.cn, yanhongli418@163.com,
{guojunqi,rfbie}@bnu.edu.cn

Abstract. Patient's motion recognition is quite popular in the area of healthcare and medical service nowadays. By analyzing the data from variant sensors within the network, we can estimate the activities a person does. The analyzing job is usually done by a classifier which can classify each motion into one category with similar movements. Self-Organizing Map (SOM) is a kind of algorithm that can be used to arrange data into different categories without any guidance. Decision tree is a mature tool for classification. In this paper, we propose a new kind of classification method with data from BAN called SOM-Decision Tree. Firstly, we use SOM on each of the sensor nodes to categorize motions into different classes, so that motions in different classes can be distinguished by this sensor. Secondly, a decision tree is constructed to discriminate each kind of movements from other motions. Finally, any action of the same patient can be recognized by query through the decision tree. According to our experiment, this algorithm is feasible and quite efficient.

Keywords: Motion recognition, SOM, Self-organizing Map, Decision Tree, classification, Mobile Health.

1 Introduction

Patient's state monitoring is a common task for medical staffs, for example, high-risk infants need to be observed day and night[1], and patients with hemiplegia should be paid attention to due to their reduced mobility. However, it becomes a heavy burden when the number of patients increased. Therefore, we need to employ information techniques to assist the detection of human status, such as the application of moving cameras in human motion detection[2]. As the development of the techniques of the Internet of things, many wearable (including implantable) wireless sensors and equipments are used in monitoring human states[3,4]. With the assistance of these technologies, patients in hospital as well as those stay at home can be monitored equally. In areas of health care and fitness, many systems have been designed to recognize and evaluate human status with the application of wireless sensors[5,6]. Among these studies, Body Area Network(BAN) is widely used to acquire human motion data. The concept of BAN was first proposed by T. G.

* Corresponding author.

K. Ren et al. (Eds.): WASA 2013, LNCS 7992, pp. 255–268, 2013.

Zimmerman and defined in the wireless World Research Forum's Book of Visions as "a collection of (inter) communicating devices which are worn on the body, providing an integrated set of personalized services to the user"[7]. As the development of wireless communication technology, BAN is extended to be wireless body sensor network(WBAN)[8]. In some study, it is also defined as Body Sensor Network(BSN)[5]. It can be used to detect human motion status, such as motion recognition[9], intervention of patient activation[10].

In order to identify different human movements, classification algorithms are commonly used, such as HMM, Bayesian classifier, SVM, etc. for example, [11,12] used decision tree to detect user states, [13] compares a reference majority voting and a naive Bayesian fusion scheme with HMM algorithm in classification for activity recognition from on-body sensors, [14] used a Bayesian classifier with multivariate Gaussians to model patient's activity, [15] compared supervised and unsupervised physical activities using a hybrid classifier combining a tree structure containing a priori knowledge and artificial neural networks. Self-organizing Map(SOM) is a sheet-like artificial neural network, that cells of which become specifically tuned to various input signal patterns or classes of patterns through an unsupervised learning process[16].It can be used as a tool in pattern recognition[17-19], clustering[20-22], classification[23-25] as well as data visualization[26]. As a cluster algorithm, SOM has been widely used in many practical areas, such as machinery health monitoring[27,28], which has proved the feasibility, practicability and priority of this algorithm as a clustering algorithm. In mobile health area, there were a few attempts of using SOM, for example, detecting the turning points of human activity based on wearable sensor array data[29], [30] used accelerometer to recognize activities of children in kindergarten, and evaluated the performance using some orthodox classifiers together with SOM, [31] introduced a visualization method for activity information sharing system using Self-Organizing Map and the activity information sharing system "ALKAN2". If we label the data in the map, SOM can also be used as a classifier. For example, in [32,33], Self-Organizing Map is used as a tool to classify a person's motion into different categories, which was the base of motion classifier that used to identify the action of fall. In this paper, we propose an algorithm that uses SOM as a clustering tool on each nodes of WBAN, which sorts different motions into various categories. As SOM is applied on each individual nodes of WBAN, we can perform the algorithm concurrently if condition permits, which increases the efficiency that matters a lot as for online algorithm. Then a decision tree[34] is constructed to perform the division of all the motions based on the results of the clustering on accelerometers within WBAN. The order of the nodes within the decision tree is chosen based on greedy strategy. According to [34], this distinction is proved to be NP-complete. Thus, we can use this method to detect patients' body motion so as to monitor their status.

Contents of the paper are as follows. In section 2 we give an overview of the basic concepts of algorithms that involved in our study. Section 3 describes the proposed algorithm in detail with its novel improvements and analysis of its application in medical care area. It is followed by several qualitative and quantitative experiments that aim to prove the feasibility, high efficiency and priority of our algorithm strategy in section 4. Then the application of this strategy in human motion recognition within medical background is simply discussed in section 5. Section 6 concludes the whole study of this paper and prospects the future work.

2 Architecture of the Motion Recognition Scheme

Movement analysis is a popular topic with a wide range of applications in health care related areas, and accelerometer is the most popular sensor applied in motion detection applications[30,35]. In our study, we focus on recognition of body motion with an array of accelerometers that form a WBAN. We design a motion recognition scheme for medical purpose which can identify various motion types of patients being observed. This scheme concentrates on discrimination of different motion types within a short period of time so that it can be applied as an online solution for medical health monitoring.

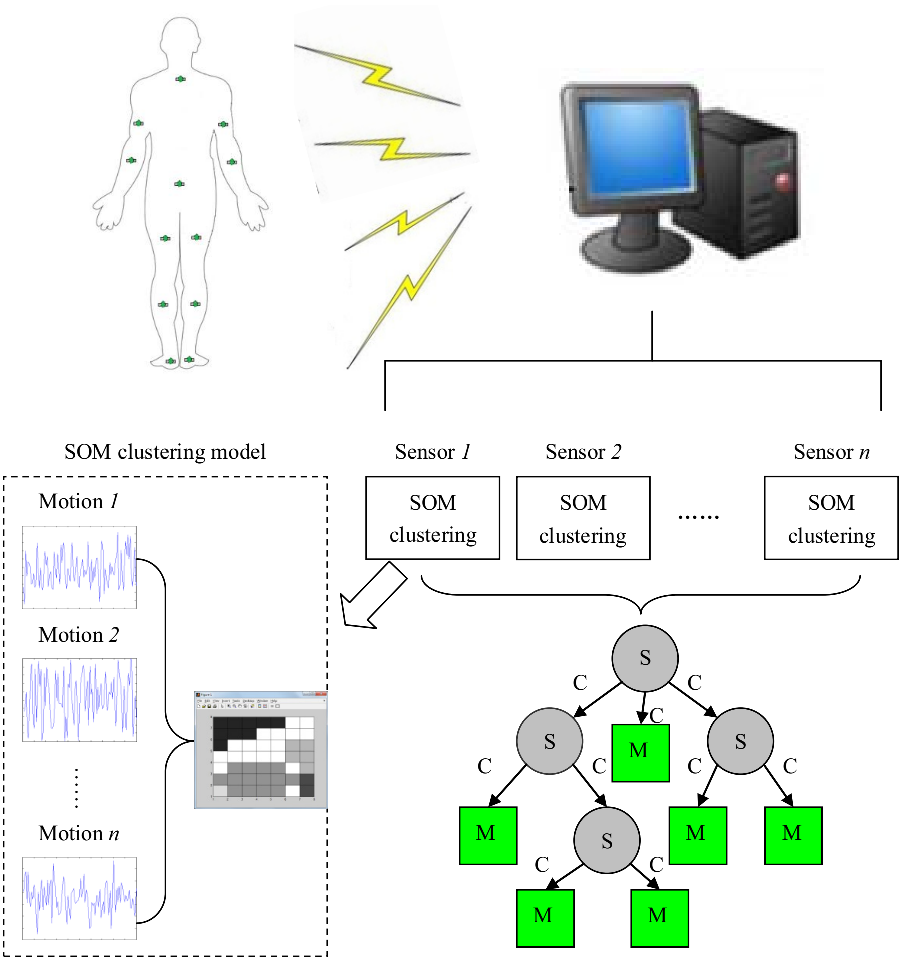

Fig. 1. Structure chart of the scheme

Our motion recognition scheme intend to be applied as a solution that employs several acceleration sensors which possess three dimensions *x*, *y* and *z*-axes along three directions as nodes of the body area network. So we use several smart phones with accelerometers to simulate motion data of sensors within the BAN refer to [32]. Data from the nodes are processed separately so as to distinguish different motion effects on the same body part. Inspired by the former study[29], we decided to analyze each nodes independently, and then choose a few with high distinguish degree with greedy policy. The structure chart of the scheme is shown in Figure 1.

The whole process of our strategy can be summarized as follow.

Data Simulation: According to different aims, different ranges of sensors should be chosen. We place 4 smart phones on different part of human body, like arm, leg, hand as 4 nodes for recognition of 8 normal actions (jogging, walking, walking down the stairs, climbing the stairs, fall, stand-sit-stand, squatting down, lying down). As the limitation of sensors in smart phones, a few simulation is done before the preprocessing. The motion data of walking, stand-sit-stand and fall acquired from the smart phone is shown in Figure 2.

Fig. 2. Motion data of walking, sitting and fall acquired from smart phone

Data Preprocessing: Data from various sensors are preprocessed for further clustering and classification. According to [32], we transform the three dimensional data array (a_x, a_y, a_z) into (a_i, t_i) array which indicates the registration of the sensor at time stamp t_i. Here, a_i can be calculated as below:

$$a = \sqrt{a_x^2 + a_y^2 + a_z^2} \tag{1}$$

In order to extract the detailed feature, we pick 30 time stamps within 3 seconds to form a data vector that represent a certain motion process in given time interval.

Data Clustering on Each Node: By dividing different motions into various groups based on their diversities on each single node, we can initially distinguish a few movements on one node, while some motions appear to be similar to other motions on certain nodes. Here in this case, we use SOM as the clustering algorithm to categorize motions on each node. As the clustering processes within each node are independent to each other, so we can perform them concurrently of the processing element of the server permits.

Motion Classification: Results of the clustering are input for further discrimination. The aim of the classification is to identify the motions within the same cluster separately from other nodes, so that all motions can be recognized. Here, we use decision tree to do the classification, and the nodes are selected using greedy strategy which has been proved to be practical in [29].

Motion Identification: After the construction of the decision tree, the classifier is ready to work. New motions can be recognized by our scheme.

3 Data Analysis Strategy and SOM-Decision Tree

Our strategy of identifying different motions by arranging them into various defined categories is to distinguish different motions roughly on each sensors of the BAN using SOM and synthesize the clustering results by discriminates motions in the same clusters using a decision tree. We describe the whole algorithm as SOM-Decision Tree, which can be used as a universal scheme for motion reorganization in BAN. This algorithm is mainly composed of two parts: clustering motions into various categories on each sensor and classifying all motions by constructing a decision tree that can distinguish motions within the same category. In the first part, as the operations of clustering on each sensor are independent to each other, we can apply concurrent computation in this process if the hardware condition is available, so that it would be more efficient with short consuming time. In the second part, we can select the most effective essential sensor array that differentiates all motions, especially the motions within the same cluster. This section is going to introduce this algorithm in details within the human motion reorganization background discussed in chapter 2.

3.1 Clustering with SOM on Each Sensor

The first step of our solution is to apply SOM on each sensor concurrently and cluster motions into different categories. SOM is the abbreviation of Self-Organizing Map, which is an unsupervised learning paradigm in artificial neural network[12]. Its main idea is to find the Best Matching Unit (BMU) and update the neural network, which is done by mapping high dimensional input vectors to analogical neurons, and modifying the weights of the best matching neuron as well as its neighbors according to their distance to the hit neuron center. After certain amount of iterations of similar training, the output neural network is ready to distinguish different kinds of input vectors, and cluster them into different categories. Here, the similarity between input data and neurons are scaled with space ranging methods, such as Euclidean distance, which is the most common way being used.

The process of the algorithm is consisted of two major steps: competition stage and cooperation stage. The first stage is also called self-organizing step, or ordering step. In this stage, the random output neurons compete with each other to become the best matching unit, which can also be treated as an ordering process for output data to be organized as the distribution of the input data. The nearest neuron of the input vector is chosen to be the BMU in each iteration, which can be understood as the neuron wins the competition with other neurons in matching with the input vector. After that, the weights of some neurons are modified in the second stage, which is also called the convergence stage. In this step, the winner has the right to modify its weight so as to be closer to the future cluster center, and its neighbors can also change their weights to some extents according to their distance to the BMU. This job is done by calculating the weight adjustment function selected by users themselves.

Concretely, the procedure of SOM algorithm can be described as follow:

1. Assigning small random numbers to the weight vectors of the output layer and performing normalization. After that, the output vector is obtained as $w_j = [w_{j1}, w_{j2}, \ldots, w_{jn}]^T$, $j = 1,2,\ldots,l$, l is the number of output neurons. The input pattern after normalization can be indicated as $x = [x_1, x_2, \ldots, x_n]^T$, n is the number of sensors.
2. Competition stage: Finding the Best Matching Unit (BMU) c using Euclidean distance method.

$$c = \arg \min_j \{\|x - w_j\|\} \tag{2}$$

Here, the Euclidean distance is calculated as:

$$\|x - w_j\| = sqrt\left[\sum_{i=1}^{n}(x_i - w_{ji})^2\right] \tag{3}$$

3. Cooperation stage: Modify the weight value of the BMU and its neighbors. The weight adjustment function is showed as follow:

$$w_j(t+1) = w_j(t) + \alpha(t)h_{cj}(t)(x - w_j(t)) \tag{4}$$

Here, $\alpha(t)$ is the learning rate that can be defined as below (α_0 is the initial value of learning rate, and τ_1 is the exponential decay function of time):

$$\alpha(t) = \alpha_0 \exp\left(-\frac{t}{\tau_1}\right) \tag{5}$$

$h_{cj}(t)$ is the neighborhood function that reveals the distance between BMU and current neuron. We decide to use the function shown in formula 6 which has been commonly used before and appears to be effective.

$$h_{cj}(t) = \exp\left(-\frac{d_{c,j}^2}{2\sigma(t)^2}\right) \tag{6}$$

Here, the distance $d_{c,j}$ is also calculated by Euclidean distance, and the neighborhood radius can be calculated using formula 7 that σ_0 is the initial value of the neighborhood radius, and τ_2 is another exponential decay function of time.

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\tau_2}\right) \tag{7}$$

It is easy to notice that both of the learning rate function and neighborhood function are annealing function.

4. If stable feature mapping is formed, then our learning is finished, otherwise, we should return to step 2 after setting $t = t + 1$.

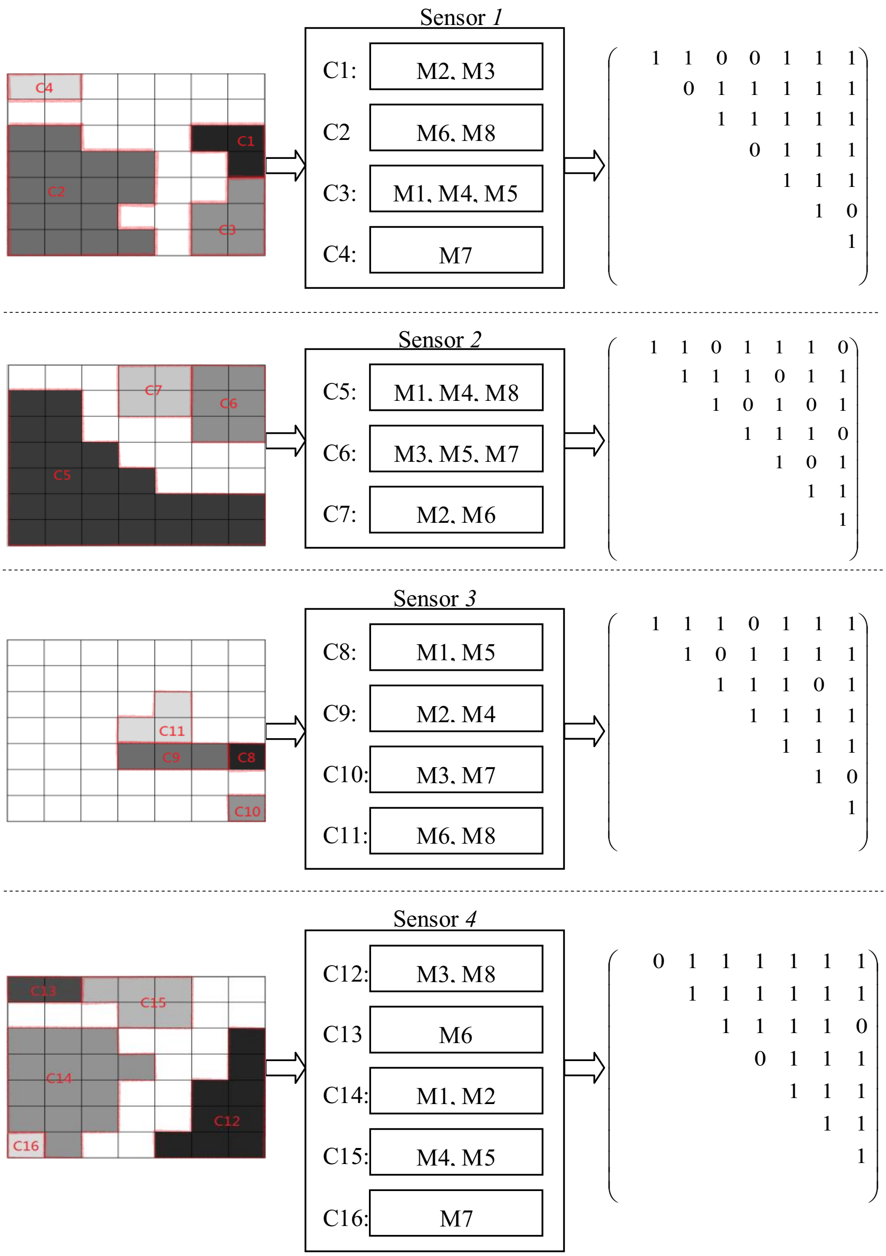

Fig. 3. Results of the SOM clustering and discrimination matrixes

As mentioned before, we select 4 nodes to distinguish 8 motions. Data from different sensors are captured and calculated using formula 1. After standardization, they will be the input vectors of the SOM clustering algorithm. The result of this step is showed in Figure 3.Here, the cells filled with the same color are considered to be in the same category. In order to check whether the current sensors are adequate to discriminate all motions, as well as determine the order of sensors in the decision tree, we construct a bunch of right upper triangular matrixes to indicate the discrimination situation of each sensor. These matrixes are easily built by finding out all the motion pairs within the same cluster and setting the corresponding elements of the matrix to 0, and the rest of the matrix elements are simply assigned with certain values according to their locations following the rules described in the previous section. The discrimination matrixes are also showed in figure 3.

3.2 Construction of the Decision Tree

The second step of our strategy is to construct a decision tree based on the clustering results of each node as well as the discrimination relations between motions represented by respective upper triangular adjacency matrix called discrimination matrix of each sensor. The possibility of the discrimination between all motions we discussed is assumed to be without doubt in [29]. So they focus on finding the complete ordering. However, here in our study, we decide to choose the essential sensors from all sensors in our BAN, and prove the distinguish ability using the complete discrimination criterion defined later in this section. If the existing sensors are proved to be adequate to distinguish all the input motions, then we can start our next step of decision tree construction. Otherwise, we might need to remind user to add more sensors to differentiate current motions. In this way, we don't need to configure too many sensors at first, as sensor number can be supervised by a few pre-experiments.

In [30], the discrimination relations are described by calculating the distinguished pairs of actions (LDS_i) on each node. However, we find it costs too much to calculate all the LDS_i, and counting the pairs of actions that can not be differentiated is easier with the same effect as the total number of relations between motions are equal in all sensors. So we choose to search indistinguishable pairs of actions by looking into the categories, as motions in the same category can not be distinguished. Other than these pairs, all the motions pairs can be treated as discriminated. Therefore, we propose a kind of right upper triangular matrix to describe the discriminated relation between motions on a certain sensor, and we called it the discrimination matrix of the sensor. Here, $w_v(i, j)$ indicates the discrimination situation between motion i and j on sensor v, that 1 indicates distinguishable, and 0 indicates indistinguishable. In this matrix, we only need to care about the right upper triangular part with $i > j$, as the matrix is symmetrical. Go through all the clusters within sensor v, we can set $w_v(i, j) = 0$, if motion i and j are within the same cluster. After filling out all the 0s, we can give 1 to the rest elements in the upper triangular part of the matrix. The discrimination matrixes of all sensors are organized to form a three-dimensional matrix W with $W(v, i, j)$ indicates the discriminated relation between i^{th} and j^{th} motions on v^{th} sensor. The establishment of matrix W is equal to the calculation of GDS and all the *LDS*. In order to ensure the number of our sensors is adequate enough to discriminate all the motions, a calculation should be done first, and our proof of the distinguish ability of our BSN is performed at the same time. This verification is done by consulting of the complete discrimination criterion below:

Complete Discrimination Criterion: Let $W' = w_1 + w_2 + \ldots + w_r$ (r represents the number of sensors), if there exists any 0 in the upper right triangular part of W', then this sensor array is not adequate to discriminate all the motions, otherwise, the sensor array is enough to distinguish the provided motions.

After the construction of W and the proof of the distinguish ability of the sensor network, a decision tree is constructed based on matrix W . Decision tree is a tree type data structure that contains two kinds of nodes: sensor node and motion node. In a decision tree, a sensor node can be either a root node or an internal node, and motion node must be a leaf node. Sensor nodes are parent nodes that possess links to child nodes, and the number of its child nodes is equal to the number of its categories. In order to unify the structure of all sensor nodes, we set $n-1$ links in its structure, because the maximum number of clusters on a sensor is n . That is to say, there might be vain links in a sensor node structure, and the number of valuable links is determined by the category number. Motion node is a leaf node without any child nodes. They are the end of the decision making process which shows the judgment of a given unknown action.

The core process of the construction of decision tree is the generation of branch nodes, especially the production of sensor nodes, which is done by choosing the most distinguishable sensor node for each branch to discriminate the motions within the corresponding category. Here, we can check the sensor matrix value of the involving motions of its parent node to estimate whether a sensor is distinguishable on certain branch. If the discrimination situation between motions in parent node shows any variation from ancient nodes, which means there exists transmission from 0 to 1, then the sensor is distinguishable, otherwise, we should choose the next node from matrix O instead of current node, as it can not distinguish motions within this category.

We use a greedy strategy to construct the decision tree, the details of which are showed as below:

Inputs: A, C, S //A: motion set; C: category set; S: sensor set
Output: T //T: the decision tree

1. For each sensor s_k , build the discrimination matrix w_v as the sub-matrix of matrix W , which meets the following conditions (Here, c_x indicates cluster on sensor v):

$$w_v(i,j) = \begin{cases} 1 & (i > j \wedge (i \in c_x \rightarrow j \notin c_x)) \\ 0 & (i > j \wedge (i \in c_x \rightarrow j \in c_x)) \end{cases} \tag{8}$$

The value of the upper triangular part of the adjacency matrix on one sensor is configured by finding all couples of motions that belong to the same category and set to 0, and the rest can be set to 1.

2. Calculate the number of zeros in each sensor's adjacency matrix, and arrange the sensors' serial numbers in accordance with the number of zeros in reverse order in matrix O . The one with least zeros ranks first and the one with most zeros comes in last.
3. A three-dimensional matrix W is built using the following formula:

$$W(v,i,j)=w_v \tag{9}$$

4. If the sensors satisfy the complete discrimination condition
5. then go to 7
6. else return and add more sensors
7. Build a stack E for sensor nodes that needed to be further developed.
8. $k=1$
9. Build a node S_k using the information from $w_{O(k)}$, and push it into the stack E .
10. $T=S$
11. While ($E \neq \phi$)
12. Pop an element S_k out of E
13. Find categories of sensor s_k that involving motions in current node S_k
14. for each of these category of sensor s_k
15. If this category contains only one common motion with S_k
16. Then build a motion node and link it to current node as its child node
17. Else
18. $y=k+1$
19. While node in $O(y)$ is indistinguishable
20. $y=y+1$
21. end while
22. Build a sensor node $S_{O(y)}$, and link it to current node as its child node
23. Push $S_{O(y)}$ into the stack E
24. end if
25. end for
26. end while

We use different shapes to represent them separately so as to be distinguished more clearly in the tree. Using the output of the first step as the input of our algorithm, the decision tree of the example we are talking about is showed as below (in Figure 4).

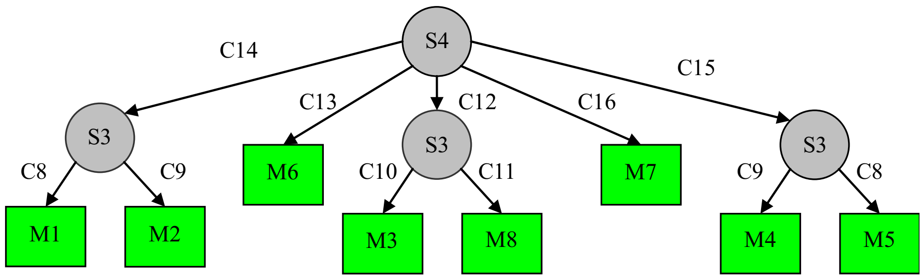

Fig. 4. The decision tree of our example

4 Simulation Analyses and Result Discussion

In order to test the feasibility and priority of the algorithm we proposed, a few simulation and experiments are done. We will compare our algorithm with two relative algorithms which have already been widely used: pure SOM and K-means-Decision tree. In our experiment, we simulate 4 sensors to distinguish ordinary motions. We use these sensors to discriminate 8 kinds of daily actions and compare the accuracy of the three methods. The result is showed in Figure 5. We find that the accuracy of

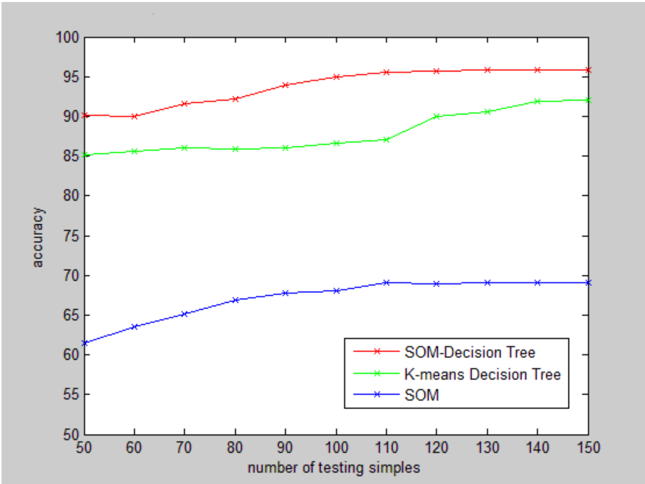

Fig. 5. The comparison of the accuracy of three algorithms

SOM-Decision Tree is the highest, as SOM can discover the accurate similar motion categories that implied in each sensor without any priori knowledge. Thus, the clusters we obtained can match the actual situation more closely. Although the result of K-means Decision Tree is also quite good, it is seriously affected by the selection of K.

The shown result display the fortunate circumstance of finding the right K, however, you might have chosen the wrong one and the performance would be shameful. In this situation, priori knowledge seems to be extremely important. However, we can not ensure to acquire enough information to choose the right number of clusters, so the performance will not be able to compete with our algorithm. Performance of SOM seems to be barely satisfied for its weak distinguish ability.

According to our experiment, 4 sensors are sufficient for recognition of 8 motions, so the workload is within the tolerance range. As the clustering can be done concurrently, the time consuming is better than pure SOM.

5 Conclusions and Future Work

In this paper, we propose an algorithm SOM-Decision Tree that combines SOM and Decision Tree to classify different human motions and intend to apply it in mobile health area. The feasibility and priority of this algorithm is easily proved by our experiment. In medical field, patient motion recognition is quite common, and it can be used for its short consuming time and high correctness.

In fact, this algorithm can be applied not only in patient motion identification, but also in many motion or gesture recognition fields. So our future work is to spread its application and build a system that can be applied in real scene.

Acknowledgement. This research is supported by National Natural Science Foundation of China (61171014) and the Fundamental Research Funds for the Central Universities.

References

1. Hayes, G.R., Patterson, D.J., Singh, M., Gravem, D., Rich, J., Cooper, D.: Supporting the transition from hospital to home for premature infants using integrated mobile computing and sensor support. Personal and Ubiquitous Computing, doi:10.1007/s00779-011-0402-4
2. Cutler, R., Davis, L.: Robust Real-Time Periodic Motion Detection, Analysis, and Applications. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 781–796 (2000)
3. Ståhl, O., Gambäck, B., Turunen, M., Hakulinen, J.: A Mobile Health and Fitness Companion Demonstrator. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations Session, pp. 65–68 (2009)
4. Xu, F., Qin, Z., Tan, C.C., Wang, B., Li, Q.: IMDGuard: Securing Implantable Medical Devices with the External Wearable Guardian. In: IEEE INFOCOM, pp. 1862–1870 (2011)
5. Shahriyar, R., Bari, M.F., Kundu, G., Ahamed, S.I., Akbar, M.M.: Intelligent Mobile Health Monitoring System (IMHMS). International Journal of Control and Automation 2(3), 13–28 (2009)
6. Bourouis, A., Feham, M., Bouchachia, A.: Ubiquitous Mobile Health Monitoring System for Elderly (UMHMSE). International Journal of Computer Science & Information Technology 3(3), 74–82 (2011)

7. Jones, V., van Halteren, A., Widya, I., Dokovsky, N., Koprinkov, G., Bults, R., Konstantas, D., Herzog, R.: Mobihealth: Mobile Health Services Based on Body Area Networks. In: M-Health Emerging Mobile Health Systems, pp. 219–236 (2006)
8. Ullah, S., Higgins, H., Braem, B., Latre, B., Blondia, C., Moerman, I., Saleem, S., Rahman, Z., Kwak, K.S.: A Comprehensive Survey of Wireless Body Area Networks. Journal of Medical Systems 36(3), 1065–1094 (2012)
9. Wu, C., Tseng, Y.: Data Compression by Temporal and Spatial Correlations in a Body-Area Sensor Network: A Case Study in Pilates Motion Recognition. IEEE Transactions on Mobile Computing 10(10), 1459–1472 (2011)
10. Solomon, M., Wagner, S.L., Goes, J.: Effects of a Web-Based Intervention for Adults With Chronic Conditions on Patient Activation: Online Randomized Controlled Trial. Journal of Medical Internet Research 14(1) (2012), doi:10.2196/jmir.1924
11. Wang, Y., Lin, J., Annavaram, M., Jacobson, Q.A., Hong, J., Krishnamachari, B., Sadeh, N.: A Framework of Energy Efficient Mobile Sensing for Automatic User State Recognition. In: Proceeding of MobiSys (2009)
12. Hong, Y., Kim, I., Ahn, S.C., Kim, H.: Mobile health monitoring system based on activity recognition using accelerometer. In: Simulation Modeling Practice and Theory, pp. 446–455 (2010)
13. Zappi, P., Stiefmeier, T., Farella, E., Roggen, D., Benini, L., Troster, G.: Activity recognition from on-body sensors by classifier fusion: sensor scalability and robustness. In: Proceeding of 3rd International Conference on Intelligent Sensors, Sensor Networks and Information, pp. 281–286 (2007)
14. Aziz, O., Atallah, B.L., ElHelw, M., Wang, L., Yang, G.Z., Darzi, A.: A Pervasive Body Sensor Network for Measuring Postoperative Recovery at Home. Surgical Innovation 14(2), 83–90 (2007)
15. Ermes, M., Parkka, J., Mantyjarvi, J., Korhonen, I.: The ingestible telemetric body core temperature sensor in Controlled and Uncontrolled Conditions. IEEE Transactions on Information Technology in Biomedicine 12(1), 20–26 (2008)
16. Kohonen, T.: The Self-Organizing Map. Proceedings of The IEEE 78(9), 1464–1480 (1990)
17. Chi, Z., Wu, J., Yan, H.: Handwritten numeral recognition using self-organizing maps and fuzzy rules. Pattern Recognition 28(1), 59–66 (1995)
18. Kitakyushu: SOM of SOMs. Neural Networks 22(4), 463–478 (2009)
19. Hu, W., Xie, D., Tan, T., Maybank, S.: Learning Activity Patterns Using Fuzzy Self-Organizing Neural Network. IEEE Transactions on Systems, Man, and Cybernetics–Part B: Cybernetics 34(3), 1618–1626 (2004)
20. Pakkanen, J., Iivarinen, J., Oja, E.: The Evolving Tree – Analysis and Applications. IEEE Transactions on Neural Networks 17(3) (2006)
21. Vesanto, J., Alhoniemi, E.: Clustering of the Self-Organizing Map. IEEE Transactions on Neural Networks 11(3), 586–600 (2000)
22. Brugger, D., Bogdan, M., Rosenstiel, W.: Automatic Cluster Detection in Kohonen's SOM. IEEE Transactions on Neural Networks 19(3), 442–459 (2008)
23. Lau, K.W., Yin, H., Hubbard, S.: Kernel Self-Organsing Maps for Classification. Neurocomputing 69, 2033–2040 (2006)
24. Suganthan, P.N.: Hierarchical Overlapped SOM's for Pattern Classification. IEEE Transactions on Neural Networks 10(1), 193–196 (1999)
25. Li, Z., Eastman, J.R.: The Nature and Classification of Unlabelled Neurons in the Use of Kohonen's Self-Organizing Map for Supervised Classification. Transactions in GIS 10(4), 599–613 (2006)

26. Vesanto, J.: SOM-based Data Visualization Methods. Intelligent Data Analysis 3(2), 111–126 (1999)
27. Côme, E., Cottrell, M., Verleysen, M., Lacaille, J.: Aircraft Engine Health Monitoring Using Self-Organizing Maps. In: Perner, P. (ed.) ICDM 2010. LNCS (LNAI), vol. 6171, pp. 405–417. Springer, Heidelberg (2010)
28. Sirola, M., Lampi, G., Parviainen, J.: SOM Based Decision Support in Failure Management. In: IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, pp. 468–473 (2005)
29. Krause, A., Smailagic, A., Siewiorek, D.P.: Context-Aware Mobile Comupting: Learning Context-Dependent Personal Preferences from a Wearable Sensor Array. IEEE Transactions on Mobile Computing 5(2), 113–128 (2006)
30. Suzuki, S., Mitsukura, Y., Igarashi, H., Kobayashi, H., Harashima, F.: Activity recognition for children using self-organizing map. In: 2012 IEEE RO-MAN, pp. 653–658 (2012)
31. Hattori, Y., Kyushu, K., Inoue, S., Hirakawa, G.: Visualization for Activity Information Sharing System Using Self-Organizing Map. In: Proceeding of International Conference in Broadband and Wireless Computing, Communication and Applications (BWCCA), pp. 537–542 (2011)
32. Kurdthongmee, W.: A Self Organizing Map Based Motion Classifier with an Extension to Fall Detection Problem and Its Implementation on a Smartphone. Applications of Self-Organizing Maps (2012)
33. Seiffert, U.: Growing multi-dimensional self-organizing maps for motion detection. Self-Organizing Neural Networks (2002)
34. Ghasemzadeh, H., Barnes, J., Guenterberg, E., Jafari, R.: A Phonological Expression for Physical Movement Monitoring in Body Sensor Networks. In: Proceeding of 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, pp. 58–68 (2008)
35. Bao, L., Intille, S.S.: Activity Recognition from User-Annotated Acceleration Data. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 1–17. Springer, Heidelberg (2004), doi:10.1007/978-3-540-24646-6_1