J Electr Eng Technol.2017; 13(?): 1921-718
http://doi.org/10.???/JEET.2017.13.1.1921

ISSN(Print) 1975-0102
ISSN(Online) 2093-7423

# Spatio-Temporal Analysis of Trajectory for Pedestrian Activity Recognition

**Young-Nam Kim*, Jin-Hee Park* and Moon-Hyun Kim†**

**Abstract** – Recently, researches on automatic recognition of human activities have been actively carried out with the emergence of various intelligent systems. Since a large amount of visual data can be secured through Closed Circuit Television, it is required to recognize human behavior in a dynamic situation rather than a static situation. In this paper, we propose new intelligent human activity recognition model using the trajectory information extracted from the video sequence. The proposed model consists of three steps: segmentation and partitioning of trajectory step, feature extraction step, and behavioral learning step. First, the entire trajectory is fuzzy partitioned according to the motion characteristics, and then temporal features and spatial features are extracted. Using the extracted features, four pedestrian behaviors were modeled by decision tree learning algorithm and performance evaluation was performed. The experiments in this paper were conducted using Caviar data sets. Experimental results show that trajectory provides good activity recognition accuracy by extracting instantaneous property and distinctive regional property.

**Keywords**: Decision tree, Fuzzy partitioning, Human activity recognition, Spatial feature, Trajectory



## 1. Introduction

Recognizing human activities is at the heart of user interfaces and applications for smart environments. It is also applied to various technologies such as surveillance security, human-computer interaction, content-based video retrieval, and virtual reality. In particular, automatic recognition of human activities is one of the most important research tasks in the field of machine learning and computer vision. And visual data is one of the most important clues in the development system for understanding and applying human behavior correctly [1, 2].

Automatic recognition of human activities requires two processes [3]: i) extraction of activity information, and ii) activity pattern modeling. Activity information refers to the movement attributes (speed, direction, location, etc.) of data and it is necessary to extract effective features for classifying human activities showing various behavior patterns. Activity patterns are expressions of frequently occurring events [4] and should be accompanied by appropriate machine learning algorithms to learn human activity patterns.

Naftel et al. [5] introduced a classification methodology for anomaly detection. His theory is related with the utilization of unsupervised learning in coefficient feature space for spatiotemporal object's trajectories. Learning trajectory patterns with Self-Organizing Maps well classifies abnormality from trajectories. But, it only detects in regular continuous activities which does not include object's static moving area due to his experimental dataset of multiple trajectories. Ce Li et al. [6] proposed an approach of detecting abnormal behavior. Her proposed methodology is based on trajectory sparse reconstruction analysis. However, it has a disadvantage that the detection performance strongly affected by Control Point Parameter which is used for Sparse Reconstruction Analysis. Histogram of optical flow is represented as a graph and graph attributes are extracted to be used as features for crowd activity in an image sequence [7]. Recently intelligent tracking algorithm computes clear trajectory of pedestrian [8] from image sequence, and there are many activity recognition studies based on trajectory. [9] However, human activity consists of several different primitive actions [10] such as slow down to put a bag, and moving opposite direction after browsing. It is required to extract regional characteristics of the trajectory depending on primitive actions in addition to sequential characteristics of time series aspect of trajectory.

In this paper, we propose an intelligent activity recognition model using the trajectory information extracted from the video sequence. The proposed model detects the characteristics of the specific trajectory region by performing each trajectory's clustering. We define it as a spatial feature. And this provides better performance in classifying different human behaviors with similar trajectory region.

This paper consists as follows. Section 2.1 describes the trajectory segmentation step. In this step, we explain segmenting a trajectory so that each segment is a part of a trajectory for a second. From the segmented trajectory,

† Corresponding Author: College of Software, Sungkyunkwan University, Korea. (mhkim@skku.edu)
* College of Software, Sungkyunkwan University, Korea. ({hwarangjin, joshdev}@skku.edu)
Received: March 21, 2017; Accepted: October 24, 2017

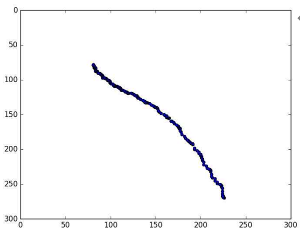
temporal features are extracted. Section 2.2 presents the fuzzy partitioning of the trajectory [11]. In this section, vector transformation of the trajectory and clustering of similar movement vectors to partition the trajectory according to moving characteristics are explained together. Section 2.3 explains feature extraction process. Feature set is divided into temporal features and spatial features, and how to compute each feature is explained. Section 2.4 shows description of the ID3 algorithm to generate classification rules as a decision tree based on the entropy of the training data set. Section 3 describes the experimental results and performance evaluation of the proposed method. We show the recognition results of four kinds of human activity modeling by comparing decision trees which are modeled with temporal features and improved decision trees by combining spatial features.

## 2. Proposed Method

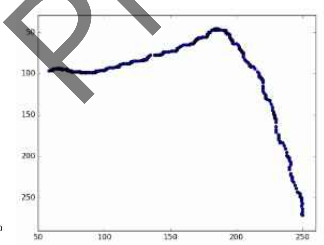
We defined four activities as shown in Table 1 for pedestrian behavior recognition research. The four activity trajectories are constructed based on GT data from the CAVIAR dataset [12].

**Table 1.** Pedestrian activity class

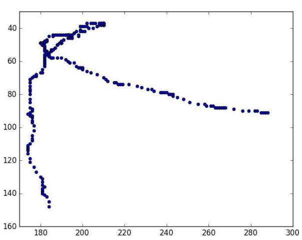
| Id | # Frames | Activity | Description |
|---|---|---|---|
| 1 | 225 | Walking | Pedestrian moving with normal stride |
| 2 | 400 | Browsing | Pedestrian moving to the desk and reading a paper for a while |
| 3 | 376 | Left bag | Pedestrian left a bag on the floor and moving |
| 4 | 500 | Fall down | Pedestrian falling down suddenly and standing up |



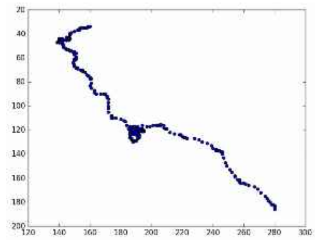
(a) Walking



(b) Browsing



(c) Left bag



(d) Fall down

**Fig. 1.** Trajectories of activity classes

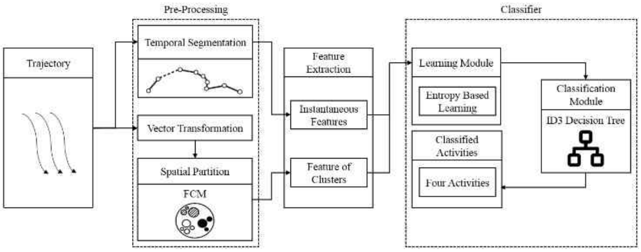



**Fig. 2.** Block diagram of the proposed method

The trajectory is a sequence of center points of the target pedestrian in each frame, $T = ((x_0, y_0), \ldots, ((x_n, y_n))$ where $(x_i,\ y_i)$ is a center position of pedestrian at ith frame. Fig. 1 shows trajectories for four behavior classes, where center position of pedestrian is represented as a dot on a two-dimensional plane. The coordinate system is expressed based on the screen coordinates as shown in Fig. 1.

In this paper, we propose a three-step pedestrian behavior recognition model as shown in Fig. 2. In the first preprocessing step, i) the pedestrian trajectory is segmented so that each segment is a displacement vector in a second, and ii) trajectory is spatially partitioned according to the similarity of displacement vector between adjacent frames. In the second feature extraction step, eight features for pedestrian behavior recognition are extracted from segmented trajectory and partitioned trajectory. The temporal features such as maximum/minimum instantaneous speed are extracted from segmented trajectory. On the other hand, the spatial features such as motional characteristics depending on the region are extracted from the partitioned trajectory [13]. Finally, in the learning and classification stage, the decision tree is learned and generated by the ID3 algorithm using extracted features. Then, when the new pedestrian trajectory data is applied, classification is performed using the generated decision tree.

### 2.1 Trajectory segmentation

This section describes the trajectory segmentation step. The trajectory is a sequence of center points of the target pedestrian in each frame, $T = ((x_0, y_0), \ldots, ((x_n, y_n))$ where $(x_i, y_i)$ is a center position of pedestrian at ith frame. When the starting point of the trajectory is $(x_0, y_0)$ and the end point of the trajectory is $(x_n,\ y_n)$, any two adjacent points in the trajectory can be expressed as $(x_m, y_m)$ and $(x_{m+1}, y_{m+1})$. Trajectory is segmented by sub-sampling point sequence at every 1 second to get segmented trajectory, $T_s = \left(\left(x(1), y(1)\right), \ldots,\ \left(x(K), y(K)\right)\right)$, where $(x(i), y(i))$ is the position of pedestrian at $i$ second, and $K$ is the length of image sequence in seconds. The video sequence used in this paper consists of 25 frames per second.

In the case of a moving pedestrian, since the density of the pedestrian center points constituting the trajectory is high, it is difficult to relate displacement of the pedestrian center points [14] between consecutive frames to physical motion of pedestrian. On the other hand, if the trajectory is divided by the second, the number of the center point of the pedestrian constituting one trajectory is reduced to 1/25, so that the density of the pedestrian central points constituting the trajectory can be greatly reduced. Considering the stride of a typical adult pedestrian, it is easy to grasp the time series characteristics of the trajectory by dividing the trajectory in seconds.

### 2.2 Fuzzy partitioning of trajectory

In this section, we explain the fuzzy partitioning method that clusters velocity vectors of trajectory into several groups and derives cluster centers [15]. To represent the spatial characteristics of the trajectory, the motional characteristics depending on the region should be extracted from the trajectory. Each velocity vector belongs to one or more groups. The membership of a velocity vector for each group is computed using heuristic measure [16].

We formulate trajectory partitioning problem as follows. Firstly, each trajectory is transformed to a vector sequence $(\overrightarrow{w_1},\ldots,\overrightarrow{w_n})$ , where each vector is a velocity vector between consecutive frames. A vector $\overrightarrow{w_i}$ is computed from $i$ th trajectory point $(x_i, y_i)$ and ( $i-1$ ) trajectory point $(x_{i-1}, y_{i-1})$ and represented in polar coordinates.

$$\overrightarrow{w_i} = [\delta_i, \theta_i],\ i = 1,\ldots, n \tag{1}$$

$$\delta_i = (x_i, y_i) - (x_{i-1}, y_{i-1}),$$

$$\theta_i = arctan\frac{v_i \cdot y}{v_i \cdot x} \tag{2}$$

Let membership value $\mu_{ik}$ be confidence degree that $i$ th vector $\overrightarrow{w_i}$ belongs to $k$th cluster. The following constraints should be satisfied by the confidence degree.

$$\sum_k \mu_{ik} = 1,\ \forall i,\ \mu_{ik} = [0, 1],$$
$$1 \le i \le n,\ \ 1 \le k \le c \tag{3}$$
$n$ : number of input
$c$ : number of clusters

Where ‘ $n$ ’ and ‘ $c$ ’ are the number of input vectors and cluster centers respectively. We are able to define the fuzzy objective function Z.

$$\text{Minimize}\ \ Z(\mu, \mathfrak{V}) = \sum_{i,\mathfrak{k}} (\mu_{ik})^p \left\| \vec{w}_i' - C_k \right\|^2 \tag{4}$$

$C_k$ : cluster center of $k$ th cluster

Where $p$ is a parameter to control fuzziness [17], and $\vec{w}_i'$ is a modified input vector to adjust relative importance of the magnitude $v_i$ with respect to orientation $\theta_i.\vec{w}_i' = [\rho\delta_i, (1-\rho)\theta_i],\ \ 1 \le \rho \le 1$ where $\rho$ is a weight of magnitude. We formulate the following Lagrange function that is the summation of objective function Z and the constraint of Eq. (5) with Lagrange multiplier $\lambda_i$ .

$$L = \sum_{i,\mathfrak{k}} (\mu_{ik})^p \left\| \vec{w}_i' - C_k \right\|^2 + \sum_i \lambda_i \left( \sum_k \mu_{ik} - 1 \right) \tag{5}$$

To derive cluster center and confidence degree, we differentiate the Language function Z with respect to $C_k$ for a fixed $\mu_{ik}$ , $\forall i$ , and differentiate $\mu_{ik}$ for fixed $C_k$ and applying normalization constraints [18]. The derived cluster center is computed as iteratively as follows: Firstly, cluster center is computed as

$$\mathrm{C_k} = \frac{1}{\sum_i^n (\mu_{ik})^p} \sum_i^n (\mu_{ik})^p\ \vec{w}_i',\ 1 \le k \le \mathrm{c} \tag{6)}$$

Secondly, the confidence degree that $i$ th vector $\vec{w}_i$ belongs to $k$ th cluster is computed as

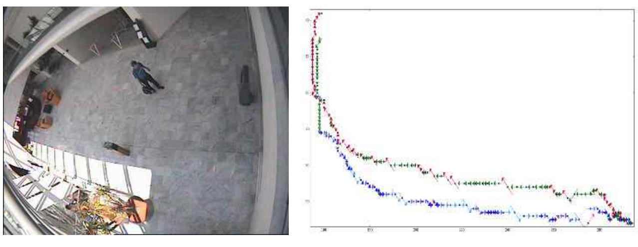
$$\mu_{ik} = \frac{1}{\sum_l^c \left( \frac{\mathrm{w_i} - \mathrm{C_k}^2}{\mathrm{w_i} - \mathrm{C}_l^{\,2}} \right)^{1/(p-1)}},\ \ 1 \le \mathrm{k} \le c \tag{7}$$

Above Eq. (6) and (7) are computed iteratively until total membership value change is small, i.e. $\sum_i \sum_k \left( \mu_{ik}^l - \mu_{ik}^{l-1} \right)^2 \le \varepsilon$ where $\mu_{ik}^l$ and $\mu_{ik}^{l-1}$ are membership values computed at lth iteration and l-literation respectively.

Fig. 3(a) shows test sequence in Caviar data set and Fig. 3(b) shows partitioned trajectory, where each vector is represented in one of four colors (red, green, blue, yellow), and each color specifies one cluster of which membership value for the vector is maximum. The membership value is represented as the saturation of the color.

### 2.3 Feature extraction

Temporal features, $f_T$ represent velocity variation characteristics with respect to time. Temporal features are



(a) Test sequence (b) Partitioned trajectory

**Fig. 3.** Trajectory Partitioning

**Table 2.** Temporal features and spatial features

| # | Name | Definition |
|---|---|---|
| | *Temporal feature* | |
| 1 | Maximum instantaneous speed ($s_M$) | $\max_{t\in[1,n]} \lvert s(t) \rvert$ |
| 2 | Minimum instantaneous speed ($s_m$) | $\min_{t\in[1,n]} \lvert s(t) \rvert$ |
| 3 | Maximum instantaneous direction change ($d_M$) | $\max_{t\in[1,n]} \theta^t - \theta^{t-1}$ |
| 4 | Minimum instantaneous direction change ($d_m$) | $\min_{t\in[1,n]} \theta^t - \theta^{t-1}$ |
| | *Spatial feature* | |
| 5 | Maximum average speed ($\gamma_M$) | $\max_{i\in[1,4]} \gamma_i$ |
| 6 | Minimum average speed ($\gamma_m$) | $\min_{i\in[1,4]} \gamma_i$ |
| 7 | Maximum average direction ($\varphi_M$) | $\min_{i\in[1,4]} \gamma_i$ |
| 8 | Minimum average direction ($\varphi_m$) | $\min_{i\in[1,4]} \varphi_i$ |

extracted from $T_s$ and Table 2 shows the temporal features defined in this paper.

Instantaneous speed, $s(t)$ shows velocity magnitude, which is moving distance of the pedestrian in 1 second.

$$s(t) = \left|\vec{v}(t)\right| = \frac{d\vec{p}}{dt} = \left(v_x(t), v_y(t)\right) \cong \left|\left(X(t) - X(t-1), Y(t) - Y(t-1)\right)\right| \tag{8}$$

If the center point of the target pedestrian at a certain time $t-1$ is defined as $\overrightarrow{P(t-1)} = \left(x(t-1), y(t-1)\right)$ and the center point of the target pedestrian at the t second is defined as $\overrightarrow{P(t)} = \left(x(t), y(t)\right)$, a line connecting the point $\overrightarrow{P(t-1)}$ and the point $\overrightarrow{P(t)}$ means a moving distance at which the pedestrian moves for one second . Thus instantaneous speed $s(t)$ is measured as Euclidean distance from the point $\overrightarrow{P(t-1)}$ to the point $\overrightarrow{P(t)}$, and it is useful for classifying the moving situation and the stagnation situation of the pedestrian.

Maximum instantaneous speed $s_M$ and minimum instantaneous speed $s_m$ show maximum instantaneous speed and minimum instantaneous speed in whole trajectory sequence, respectively.

Moving direction change, d shows difference between current moving direction of pedestrian and previous moving direction, where $\theta^t$ is moving direction of pedestrian between $t-1$ second and t second, and $\theta^{t-1}$ is previous moving direction between $t-2$ second and $t-1$ second.

$$d = \theta^t - \theta^{t-1}, \theta = tan^{-1}\left(\frac{y(t) - y(t-1)}{x(t) - x(t-1)}\right), \tag{9}$$

Moving direction change is measured to describe occurrence of distinctive action such as go back to check the bag in left bag sequence which makes sudden change of moving direction. Maximum direction change $d_M$ and minimum direction change $d_m$ is computed as maximum and minimum value for a whole trajectory, respectively.

The second feature group is spatial features $f_S$ extracted from partitioned trajectory $T_P$. The trajectory $T$ is partioned firstly, according to motion characteristics into several clusters each of which shows regional distribution of similar motion. Spatial features represent regional characteristics for each cluster [11]. For cluster $C_i$, average speed $\gamma_i$ is computed as

$$\gamma_i = \frac{\sum_j \mu_{ji}\delta_j}{\sum_j \mu_{ji}} \tag{10}$$

and average moving direction $\varphi_i$

$$\varphi_i = \frac{\sum_j \mu_{ji}\theta_j}{\sum_j \mu_{ji}} \tag{11}$$



The minimum average speed $\gamma_m$ and maximum average speed $\gamma_M$ are computed as minimum and maximum of average speed for all clusters. Also, minimum average moving direction $\varphi_m$ and maximum average moving direction $\varphi_M$ are computed as minimum and maximum of average moving direction for all clusters.

### 2.4 Decision tree

When a feature vector is generated for each of the sample trajectories, classification is performed using a decision tree. A decision-tree-based classification method [19] was selected, since the classification module is designed as a part of knowledge based system and the effective features of the tree-structure-derived logical expression are easy to be combined with inference algorithm. For each trajectory, the temporal features $f_T = \{f_{T_1}, \ldots, f_{T_n}\}$ and spatial features $f_S = \{f_{S_1}, \ldots, f_{S_n}\}$ are extracted from segmented trajectory and partitioned trajectory, respectively. The first decision tree $D_T$ is constructed from training set X described using only temporal features. The second tree $D_S$ is drived from training set X described using both temporal features and spatial features.

Each decision tree is generated from an ID3 learning algorithm which is an information-theoretical algorithm [20, 21]. Each iteration of the ID3 tries to attain the largest information gain (or smallest entropy value), and a probability of $p$ is determined from the occurrence frequency. The entropy of a given data set $T$ is computed as follows:

$$E(T) = -\sum_{C_i \in C} p(C_i) log_2 p(C_i) \tag{12}$$

where $C$ is a set of classes in $T$ that is four class of trajectories, and $p(C_i)$ is the proportion of the number of pairs labeled as class $C_i$ in $T$. The information gain(IG) from the partitioning of $T$ according to the value of the feature $f_x$ is computed using Eq. (13), as follows:

$$IG(f_x, T) = E(T) - \sum_{v_j \in V_x} p(v_j) E(T_{v_j}) \tag{13}$$

In Eq. (13), $E(T_{v_j})$ is the entropy of the subset $T_{v_j}$ that is constructed through the collection of pairs, each of which has a value of $v_j$ for the feature $f_x$, from $T$. $V_x$ denotes a set of values for $f_x$; therefore $T = \bigcup_{v_j \in V_x} T_{v_j}$. The probability $p(v_j)$ is computed using Eq. (14), as follows:

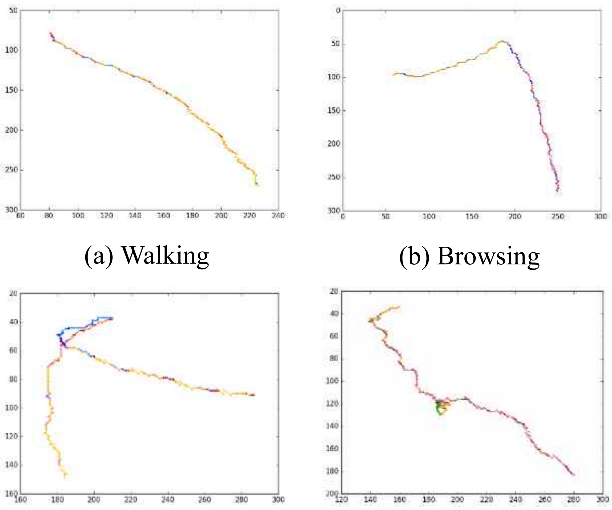
$$p(v_j) = \frac{|T_v|}{|T|} \tag{14}$$

At each iteration of the construction of a decision tree, the feature that provides the maximum information gain is selected as a decision node [22, 23]. To deal with the numeric features, the C4.5 algorithm is used [24, 25]. The learned decision tree is used in the classifier to recognize the behavior of the pedestrian. At every node of the decision tree, it makes a yes / no decision through a comparison operation and moves to the child node according to the result. This process is repeated cyclically until a leaf node is encountered.

## 3. Experimental Classification Results and Analysis

We experimented the proposed method with 'Caviar Test Case Scenarios' dataset which includes people walking alone, meeting with others, window shopping, entering and exiting shops, fighting and passing out and last, and leaving a package in a public place. In our experiment, we selected four sequences consist of people walking, fall down, leaving his bag and browsing on the floor from Caviar dataset. In the next paragraph, we will explain how to extract spatial feature from trajectory.

First of all, we conjugate frame segmentation of each trajectory and plot object's virtual trajectory on 2d image plane. Caviar Dataset offers each frame's blob center position. From this data, we generated vector transformation of the trajectory of the pedestrian for each image sequence. Fuzzy partitioning of vector sequence was executed by 'Figue' which is a collection of clustering algorithms implemented in JavaScript open sourced under MIT License [26]. It provides functions to visualize clustering output as dendograms. In the experiment, we let $\rho$ be 0.5. The number of clusters for each activity class is



(a) Walking (b) Browsing

(c) Left bag (d) Fall down

**Fig. 4.** Partitioned Trajectories

**Table 3.** Cluster Centroids

| Class | ClusterID0 ($\varphi_0$, $\gamma_0$) | ClusterID1 ($\varphi_1$, $\gamma_1$) | ClusterID2 ($\varphi_2$, $\gamma_2$) | ClusterID3 ($\varphi_3$, $\gamma_3$) |
|---|---|---|---|---|
| Walking | (-2.250, 8.109) | (-2.357, 2.832) | (-2.091, 12.222) | (-2.315, 2.772) |
| Browsing | (0.084, 0.542) | (1.032, 8.830) | (-0.060, 5.241) | (-0.091, 3.205) |
| Left bag | (-0.566, 6.246) | (-0.234, 1.567) | (-0.345, 10.231) | (1.379, 1.220) |
| Fall down | (1.027, 6.562) | (0.595, 7.080) | (0.0318, 0.404) | (0.943, 1.927) |

assigned 4 equally. From the output of clustering, we are able to obtain the membership value of a vector to each cluster and each cluster's centroid from transformed vectors. The membership value must be from 0 to 1 and the summation of these values is 1. To visualize clustering result, we used color saturation to denote membership value in Fig. 4. Different color (red, green, blue, and yellow color) is assigned to each cluster, and the color is selected as the color of the cluster which has maximum membership value for the vector.

From Fig. 4(b), the difference between moving direction before shopping and moving direction after shopping makes different clusters. Also in left-bag sequence Fig. 4(c), moving direction change around leaving the bag point makes different clusters. The partitioning result is also shown in terms of cluster's centroid for each class as in Table 3. In Table 3, the first component and second component of each centroid denote $\varphi_i$ and $\gamma_i$ respectively. The distribution of cluster centroids is shown in Fig. 5. It shows distinctive distribution for each class.

Four types of human activity pattern modeling process were modeled by combining temporal features and spatial features. The C4.5 algorithm used in the study of four human activity patterns was applied by Weka 3.6 version

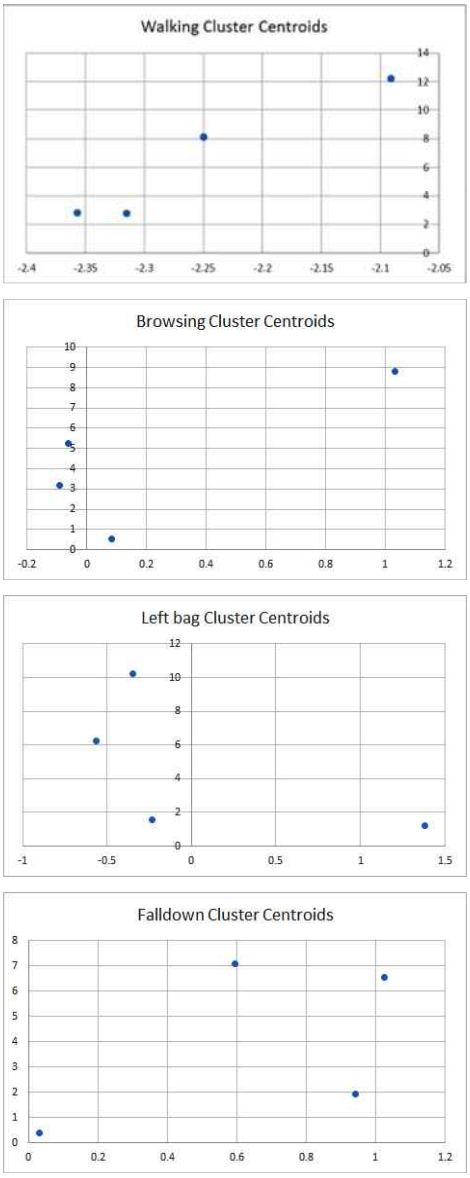



**Fig. 5.** Scatter diagram of cluster centroids

of the data mining tool. Weka is a representative open data mining tool developed by the University of Waikato in New Zealand and provides interfaces for evaluating the results of various machine learning algorithms and learning schemes for a given set of data [27].

Fig. 6 is a decision tree learned from training data using only temporal features. The number of leaf nodes is 5 and the size of the tree is 11. At depth 1, the walking class is distinguished from other classes with a minimum instantaneous speed attribute. The walking class has larger instantaneous moving speed compared with other three classes. At depth 2 and depth 3, the browsing class is distinguished from other classes by the minimum instantaneous speed attribute and the maximum instantaneous speed attribute. The browsing class reduces the movement speed of the pedestrian when the event occurs, but does not become zero. Therefore, the

**Table 4.** Confusion Matrix using decision tree ($D_t$)

| | Walking (Predicted) | Browsing (Predicted) | Leftbag (Predicted) | Falldown (Predicted) |
|---|---|---|---|---|
| Walking (Actual) | 100% | 0% | 0% | 0% |
| Browsing (Actual) | 0% | 100% | 0% | 0% |
| Leftbag (Actual) | 12.5% | 0% | 62.5% | 25% |
| Falldown (Actual) | 0% | 12.5% | 25% | 62.5% |

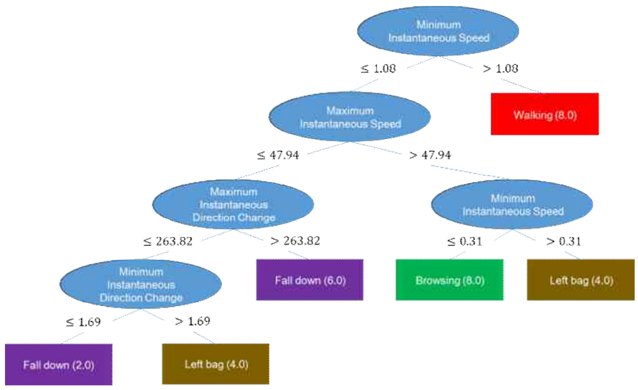



**Fig. 6.** Decision Tree ($D_t$) using temporal features

instantaneous velocity attribute effectively classifies the browsing class rather than the instantaneous direction change property. At depth 3, the maximum direction change attribute was chosen as an attribute to classify the fall down class. This indicates that the fall down class is the class with the largest change in pedestrian direction. We evaluated the performance of a decision tree constructed with 10-fold cross-validation. Table 4 shows confusion matrix of the classification result using decision tree of Fig. 6. The walking and browsing classes are classified correctly, but the true positive rate of the left bag class is 62.5% and confused with walking class. Also, the true positive rate fall down class is 62.5% and confused with browsing class.

Fig. 7. shows a decision tree generated from training data represented both temporal features and spatial features. The number of leaves is 4 and the size of the tree is 7. In the improved decision tree, there are significant changes in the four categories of human behavior. It has been divided into two groups at depth 1 based on minimum average speed property, which is very intuitive. At depth 2, the first group was classified as a browsing class and a fall down class by the maximum instantaneous speed attribute. The second group was classified as a walking class and a left bag class by the minimum direction change property. Table 5 shows the confusion matrix of classification results using new decision tree generated using temporal features and spatial features. Likewise, the performance of decision tree was evaluated by 10-fold cross-validation. The true positive rates of walking, browsing, left bag, and fall down classes are all 87.5 %. While recognition

**Table 5.** Confusion Matrix using decision tree ($D_s$)

| | Walking (Predicted) | Browsing (Predicted) | Leftbag (Predicted) | Falldown (Predicted) |
|---|---|---|---|---|
| Walking (Actual) | 87.5% | 0% | 12.5% | 0% |
| Browsing (Actual) | 12.5% | 87.5% | 0 | 0% |
| Leftbag (Actual) | 12.5% | 0% | 87.5% | 0% |
| Falldown (Actual) | 0% | 12.5% | 0% | 87.5% |

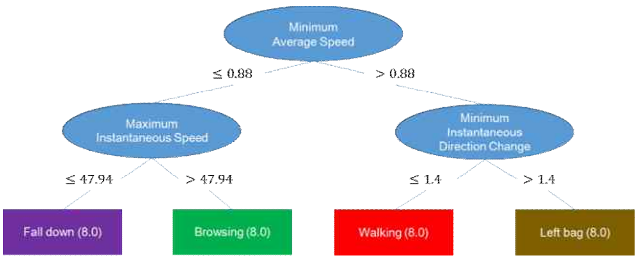



**Fig. 7.** Decision Tree ($D_s$) using spatial features and temporal features

accuracy of walking class and browsing class degraded, recognition accuracy of left bag class and fall down class is improved compared with accuracies of Table 4. The first classification using minimum average speed of decision tree does not give better accuracy for waling class and browsing class. It results from greedy search characteristics for attributes in generating decision tree. However, the overall precision rises from 81.25% of Table 4 to 87.5% of Table 5. Experimental results show that although temporal features alone have limitations in recognizing various human activities, it is possible to recognize human activities more adaptively by combining spatial features. Since there are many similar trajectory sections across classes, the distinctive trajectory section of each class can be identified by partitioning trajectory. This experiment proved that the spatial features extracted from distinctive trajectory section are effective features for recognition of activity classes which include sudden change in motion such as left bag class and fall down class.

## 4. Conclusion

We proposed a new recognition technique for pedestrian activity recognition. The proposed recognition method extracts temporal features and spatial features based on the trajectory information obtained from video data. When the decision tree is generated by only the traditional temporal features from trajectory, there is a problem not to consider regional characteristics of the trajectory to limit the activity classification precision. S we have proposed intelligent spatial feature extraction algorithm by fuzzy partitioning the trajectory. The spatial features extracted from clusters of the trajectory are used to generate new decision tree and it has been shown to be effective for classification of various human behaviors with increased recognition performance. In the future study, we will develop more temporal features and spatial features to recognize more complicated group behaviors such as fighting with other or chasing other [28].

## Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (NRF-2014R1A2A1A11053902).

## References

[1] Kim, Jinpyung, et al. “Crowd Activity Recognition using Optical Flow Orientation Distribution,” TIIS 9.8 (2015): pp. 2948-2963.
[2] A. Mehrabian and M. Wiener, “Decoding of inconsistent communications,” *J. Personality Soc. Psychol.*, vol. 6, no. 1, pp. 109-114, 1967.
[3] Turaga, Pavan, et al. “Machine recognition of human activities: A survey,” *IEEE Transactions on Circuits and Systems for Video Technology*, 18.11 (2008): pp. 1473-1488.
[4] Hu, Weiming, et al. “Learning activity patterns using fuzzy self-organizing neural network,” IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 34.3 (2004): pp. 1618-1626.
[5] Naftel, Andrew, and Shehzad Khalid. “Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space,” Multimedia Systems 12.3 (2006): pp. 227-238.
[6] Li, Ce, et al. “Visual abnormal behavior detection based on trajectory sparse reconstruction analysis,” Neurocomputing 119 (2013): pp. 94-100.
[7] Aggarwal, Jake K., and Michael S. Ryoo. “Human activity analysis: A review,” ACM Computing Surveys (CSUR) 43.3 (2011): p. 16.
[8] Boltes, Maik, et. al. “Automatic extraction of pedestrian trajectories from video recordings,” Pedestrian and Evacuation Dynamics 2008. Springer Berlin Heidelberg, 2010. pp. 43-54.
[9] Morris, Brendan Tran, and Mohan Manubhai Trivedi. “A survey of vision-based trajectory learning and analysis for surveillance,” *IEEE transactions on circuits and systems for video technology 18.8* (2008): pp. 1114-1127.
[10] Poppe, Ronald. “A survey on vision-based human action recognition,” Image and vision computing 28.6 (2010): pp. 976-990.
[11] Chuang, Keh-Shih, et al. “Fuzzy c-means clustering with spatial information for image segmentation,” computerized medical imaging and graphics 30.1

(2006): pp. 9-15.
[12] Homepages.inf.ed.ac.uk, CAVIAR Test Case Scenarios. http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/
[13] Lee, Jae-Gil, et al. “TraClass: trajectory classification using hierarchical region-based and trajectory-based clustering,” Proceedings of the VLDB Endowment 1.1 (2008): pp. 1081-1094.
[14] Lee, Seon-Woo, and Kenji Mase. “Recognition of walking behaviors for pedestrian navigation,” Control Applications, 2001.(CCA'01). Proceedings of the 2001 IEEE International Conference on. IEEE, 2001.
[15] Pelekis, Nikos, et al. “Clustering trajectories of moving objects in an uncertain world,” Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on. IEEE, 2009.
[16] Eom, Ki-Yeol, Jae-Young Jung, and Moon-Hyun Kim. “A heuristic search-based motion correspondence algorithm using fuzzy clustering,” International Journal of Control, Automation and Systems 10.3 (2012): pp. 594-602.
[17] Dombi, József. “A general class of fuzzy operators, the DeMorgan class of fuzzy operators and fuzziness measures induced by fuzzy operators,” Fuzzy sets and systems 8.2 (1982): pp. 149-163.
[18] Kim, Dae-Won, Kwang H. Lee, and Doheon Lee. “Fuzzy clustering of categorical data using fuzzy centroids,” Pattern Recognition Letters 25.11 (2004): pp. 1263-1271.
[19] Friedl, Mark A., and Carla E. Brodley. “Decision tree classification of land cover from remotely sensed data,” Remote sensing of environment 61.3 (1997): pp. 399-409.
[20] Quinlan, J. Ross. “Induction of decision trees,” Machine learning 1.1 (1986): pp. 81-106.
[21] Cheng, Jie, et al. “Improved decision trees: a generalized version of id3,” Proc. Fifth Int. Conf. Machine Learning. 1988.
[22] Michalski, Ryszard S., Jaime G. Carbonell, and Tom M. Mitchell, eds. Machine learning: An artificial intelligence approach. Springer Science & Business Media, 2013.
[23] Stein, Gary, et al. “Decision tree classifier for network intrusion detection with GA-based feature selection,” Proceedings of the 43rd annual Southeast regional conference-Volume 2. ACM, 2005.
[24] J. R. Quinlan, “C4.5: Programs for Machine Learning,” Morgan Kaufmann Publishers, San Mateo, CA, 1993
[25] Quinlan, J. Ross. “Improved use of continuous attributes in C4. 5,” Journal of artificial intelligence research 4 (1996): pp. 77-90.
[26] Google Code Archive, “Figue - A Collection of Clustering Algorithms Implemented in Javascript,” https://code.google.com/archive/p/figue/wikis/Introduction.wiki
[27] Cs.waikato.ac.nz, “Weka 3 - Data Mining with Open Source Machine Learning Software in Java,” http://www.cs.waikato.ac.nz/ml/weka/
[28] Lin, Weiyao, et al. “Group event detection with a varying number of group members for video surveillance,” IEEE Transactions on Circuits and Systems for Video Technology 20.8 (2010): pp. 1057-1067.

**Young-Nam Kim** He received the M.S. degree in College of Information and Communication Engineering from Sungkyunkwan University, Suwon, Korea, in 2015. He is currently a Ph.D. candidate in Sungkyunkwan University, Suwon, Korea. His research interests include artificial intelligence, machine learning, and computer vision.

**Jin-Hee Park** He received the B.S. degree in applied IT and Engineering from Pusan National University, Busan, Korea in 2016. He is currently a M.S candidate in Sungkyunkwan University, Suwon, Korea. His research interests include artificial intelligence, machine learning, and computer vision.

**Moon-Hyun Kim** He received the B.S. degree in Electronic Engineering from Seoul National University in 1978, the M.S. degree in Electrical Engineering from Korea Advanced Institute of Science and Technology, Korea, in 1980, and the Ph.D. degree in Computer Engineering from the University of Southern California in 1988. From 1980 to 1983, he was a Research Engineer at the Daewoo Heavy Industries Co., Seoul. He joined College of Software, Sungkyunkwan University, Seoul, Korea in 1988, where he is currently a Professor. In 1995, he was a Visiting Scientist at the IBM Almaden Research Center, San Jose, California. In 1997, he was a Visiting Professor at the Signal Processing Laboratory of Princeton University, Princeton, New Jersey. His research interests include artificial intelligence, machine learning, and pattern recognition.