



Classifying pedestrian trajectories by Machine learning using laser sensor data

Hiroyuki Kaneko¹ and Toshihiro Osaragi²

¹ Kajima Technical Research Institute, Tokyo, Japan

² School of Environment and Society, Tokyo Institute of Technology, Tokyo, Japan

Correspondence: Hiroyuki Kaneko (kaneko-hiroyuki@kajima.com)

Abstract. In the field of facility planning, the analysis of pedestrian trajectories using laser sensor-based behavior monitoring technologies is a proven way to improve our understanding of the behavioral features of foot-travelers. While these technologies can gather large volumes of trajectory data, the analysis of such data is a chaotic and complicated task and creates a large workload if it must be interpreted visually by human analysts. Hence, a method is needed for automatically extracting the features and their separate components from pedestrian trajectories and patterns. This study proposes just such a method based on a Restricted Boltzmann machine, a machine learning tool, to automatically extract and classify the latent features of pedestrian trajectories. Our method was applied to data taken in the outpatient waiting area of a hospital and the machine learning generated results were compared to those of visual classifications by human analysts. It was shown to be functional for classifying trajectories by orientation, stopping location and walking speed, and was considered effective for furnishing rough classifications resembling the intuition-based classifications of a human analyst.

Keywords. pedestrian trajectories, laser sensor, Restricted Boltzmann Machine (RBM), classification of trajectories

1 Introduction

1.1 Background of this study

It is essential for facility planners to gain a deeper understanding of the behavioral features of people as they walk, in order to create plans that better fit natural lines of human movement. Conventionally, most research has relied on follow-up investigations or survey responses, but as information and communication technologies have spread in recent years, it has become possible to use the

maturing technologies for behavioral measurement on highly precise and long-term data about the uses of and conditions in architectural spaces. Laser sensor-based behavior monitoring technologies (laser metrology) are among these measurement technologies.

Previously, the authors have used data gained by this method in office spaces in order to observe what kinds of activities occur, where, and at what time of day, and to seek ways to visualize that information to be easy to understand (H.Kaneko and T.Osaragi, 2015). For example, we have recently proposed, as a bird's-eye view of an entire space, a procedure for representing the features of spatio-temporal uses of a given space by analysing the contours of the user trajectories.

In this study, we propose a procedure for classifying the behavioral patterns of facility users by extracting latent features of the trajectories of people as they walk through the facility. For example, by looking closely at pedestrian trajectory data in the outpatient reception area of a hospital, we have been able to identify the attributes of specific individuals (e.g., patient or medical staff) by observing, for example, whether they (a) came in through the entrance, waited at the return visit reception machine, and then approached the examination area; (b) came from other buildings and proceeded briskly toward a patient wing; or (c) came from an examination area, submitted forms at the cashier's desk, and then sat in a waiting area. The aspects used for these analyses included the orientation of movement, i.e., from where to where the individual is moving, locations where the individual is immobile, such as a return visit reception machine or the cashier's desk, and the individual's walking speed, which is characteristically slow for aged persons.

Fig.1 provides a layout of the locations of the laser sensors in the outpatient reception area of a certain hospital and Fig.2 shows the results of a visual classification of 157 pedestrian trajectories. This classification effort focused mainly on the person's orientation, where he/she stopped, and whether he/she sat down. Such information could be read from the person's trajectory, but the trajectories that were actually observed were quite diverse, complicated, and even chaotic.

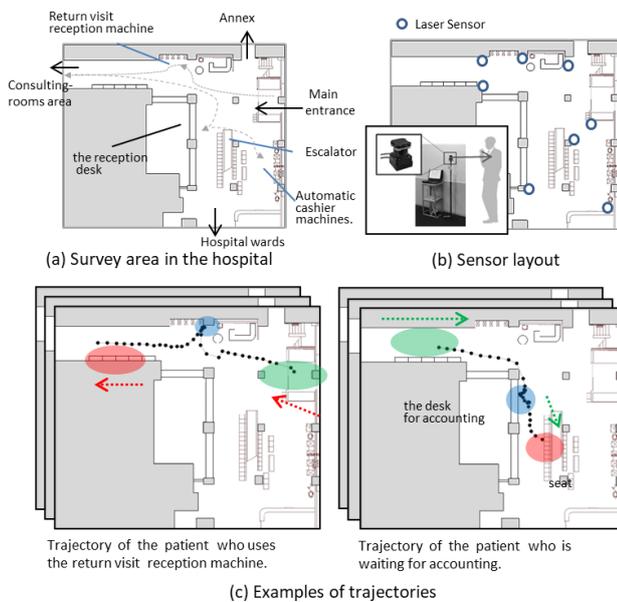


Figure 1 . Outline of pedestrian trajectory data

(a) Result of manual classification

#	Patterns from trajectories	Number
A1	Up-r side → Down side	29
A2	Up-r side → Down side: RighthRoot	13
A3	Down side → Up-r side	24
A4	Down side → Up-r side: RighthRoot	7
A5	Up-r side → Left side	9
A6	Up-r side → <RVRM> → Left side	7
A7	Lef tside → Up-r side	6
A8	Left side → Down side	9
A9	Lef tside → <Cashier's window> → take a seat	5
A10	Up-r side → <Standing for wait> → take a seat	2
A11	Down side → Left side	11
A12	Up-r side → <Escalator>	6
A13	<Office> → Down side	2
A14	Other	27
Total		157

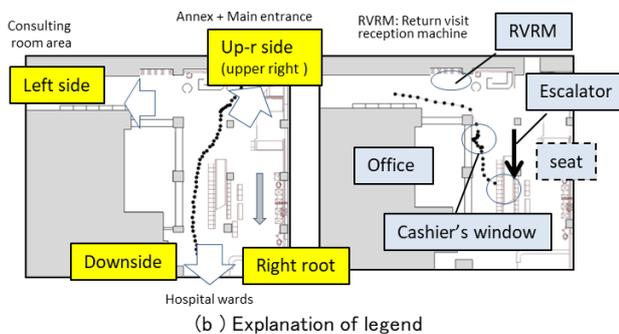


Figure2. Result of visual classification by manual

If unusual trajectories that occurred for only one or two people each can be reasonably assigned to the “other” classification, we broadly divided the remaining trajectories into 13 classifications. However, classifying a large amount of trajectory data by behavioral features of diverse individuals by eye is a lengthy and cumbersome process. Additionally, when such tasks are drawn out in time, they are susceptible to irregularities in the classification standards, as well as differing standards among individual analysts. This indicates the need for a method that can be used to automatically extract and classify the key features describing a behavioral pattern seen in a person's trajectory.

1.2 Previous research and objective of this study

The methods for classifying trajectory data can be broadly divided into supervised classification and unsupervised classification.

Supervised classifications suggest methods for machine-learning classification, such as support vector machines (SVM) and random forests, using data that analysts have already determined to classify. Papathanasopoulou (2019) tried the state classification of pedestrians (the state of being distracted by something as a cell phone.) using the pedestrian trajectory data of GNSS and the random forest. However, hand-made prior classification work is often time-consuming, so in this report, we would like to discuss unsupervised classification.

Unsupervised classifications differ in terms of data granularity. For using GPS data, if the destination is a transition of discrete type state, unsupervised clustering can be achieved by using latent class analysis that assumes a categorical latent variable behind the observation data. By this method, Tanaka (2015) tried to classify walking, bicycle, bus, car, and train as means of transportation based on the moving speed component of GPS data.

On the other hand, when laser sensor-based behavior monitoring data is used, three methods, detailed below, have been established for unsupervised classifying trajectory data:

Method(1): Time series are examined in pairs for classification by their similarities; method(2): Classification after dimensional compression of a trajectory in a linear space; and method(3): Classification after dimensional compression of a trajectory in a non-linear space.

Dynamic time warping (DTW; P.Senin, 2008) is one way to perform method (1). In this process, each of the points of two time series are compared with each other and the path providing the shortest distance between the two series is identified. Then, the trajectory group is classified with the length (DTW distance) matrix using K-means

clustering or some other method. Although this method has computational advantages even when the path lengths are different, since it compares for resemblance between all points of each path, it cannot identify which features, if any, are similar between the two paths. Other concerns are that stoppage times can be omitted, so information about stopping locations is discarded, and the method's high calculation cost.

Principal Component Analysis (PCA) provides a way to perform method (2). In the classification method using PCA for abnormality detection (Ukai, 2007), all points are linearly interpolated in advance and dimensionalized (measurement points \times 2) in order to make the data length equal. Since multidimensional data is easy to handle by dimensional compression, PCA is performed to classify trajectory data into low-dimensional spaces. This calculation process is easy but discards the walking speed information in the pre-processing step when the lengths of all data are set to a uniform value.

Character recognition using a technique for generating text from handwritten characters is a technique suitable for method (3). One such technique is a restricted Boltzmann machine (RBM; Hinton, G. E. et al., 2006), in which a specific value for a shape in the image data of a character is copied into a nonlinear space for classification as a character.

Since the trajectories are a time series data having position coordinates like characters, it is difficult to find the regularity of the point cloud connection using a clustering method such as a k-means with the distance of similarity. For this reason, it is advantageous to use a deep learning pre-training method that automatically discovers and learns the regularities. While there are deterministic pre-trained "auto-encoder", probabilistic pre-trained "RBM" is known to be more useful.

It was thought that an RBM could be applied more directly to the problem of classifying pedestrian trajectories. Hence, in this paper, a procedure for automatically extracting and classifying the latent features of trajectories in input data was investigated, and a model was constructed to represent the orientation of the patients and the locations where they stopped. A classification calculation was then carried out using trajectory data collected in an actual hospital outpatient waiting area to validate the capability of the proposed method by comparing its results with those of visual assessments.

2. Trajectory Classification Model Based on Restricted Boltzmann Machine

2.1 Overview of restricted Boltzmann machine

The RBM is a probabilistic graphical model capable of unsupervised learning of stochastic distributions in training data (C.M.Bishop,2006). It comprises a two-layer network: an input layer (visible layer) and a hidden layer. When learning proceeds such that the associated values representing the values in the hidden layer (which are generated from the data in the input layer and are returned to the original neuron in the input layer) resemble the original data as closely as possible, the patterns in the hidden layer data expressing their features in that layer are generated (Fig. 3).

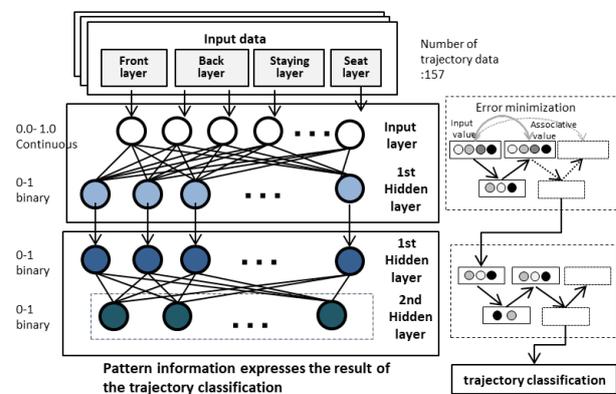


Figure 3. Trajectory classification using RBM

Stimuli from the input layer (multiplied by the weights of the input values and with a bias added) pass the activation function (the sigmoid function is used), and since a neural net generates the values to be output from the hidden layer, it is capable of non-linear classification. This task can be spread over as many levels as desired and is well known to provide representations of more complicated features. When a model contains two or more hidden layers, it is called a deep Boltzmann machine. In one approach to the learning method, teaching data, the solution, is also provided on the final layer. However, in this study, learning was "unsupervised", without correct data during learning. Instead, classes were represented by differences in the patterns in the hidden layer patterns.

2.2 Model for representing pedestrian's trajectory

In this study, we examine the problem of machine learning for classification of pedestrian trajectories in a hospital outpatient waiting area, as measured by laser metrology. The first step was viewed as an essential and fundamental capability, to automatically identify situations such as walking through the waiting area without stopping or stopping at specific locations such as the return visit

reception machine. A model was then constructed to represent orientation and stopping locations.

The lengths of trajectory data were found to be quite inconsistent with each other, primarily due to the distance and duration differences for stopping times. Therefore, since they cannot be handled as-is with trajectory IDs, times and x-y coordinates, time data were abstracted, and space data were converted to mesh data.

The data processed in this study do not contain complicated trajectories with multiple destinations, so the data were sufficient to describe the orientation of a trajectory as essentially from the entrance toward the examination area or another location. Each trajectory was divided at the midpoint into the front and back layers, which consisted of spatial meshes. When it was necessary to handle a complicated motion, the trajectory itself needed to be sectioned into partitions, but as long as the key information to be described consisted simply of “from where to where”, two partitions sufficed. A third “staying layer” was added in order to describe stopping at particular locations such as the return visit reception machine. A detailed examination of the data shows that the ends of some of the trajectories are at “sitting down/leaving the seat”. These were also recognized as significant data, and a flag was added to indicate whether the “origin” or “disappearance” of the person had occurred at a seat.

The actual procedure for creating the input data was as follows:

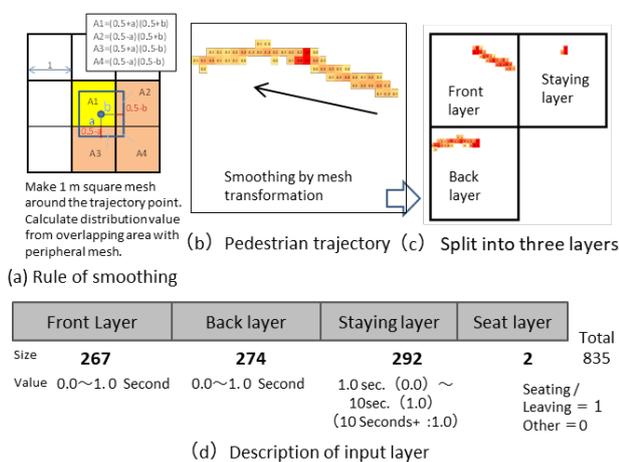


Figure 4. Pedestrian trajectory model

(1) The trajectory points were created at 1 s intervals on a 1×1 m two-dimensional mesh. The data were handled as one-dimensional in the RBM calculations, so the changes in the values were smoothed to make the trends easier to recognize. There are several methods for smoothing, but in this study, each trajectory point was mapped in a proportionately weighted fashion to the neighbouring mesh points

falling within the 1×1 m square centred on the trajectory point (Fig.4-a,b). Fig.5 shows data distributions of walking speeds with and without smoothing. The reader can see that there is a risk that walking speeds, which are inherently continuous, are not correctly represented when handled in a discrete manner while location information is unprocessed for smoothing.

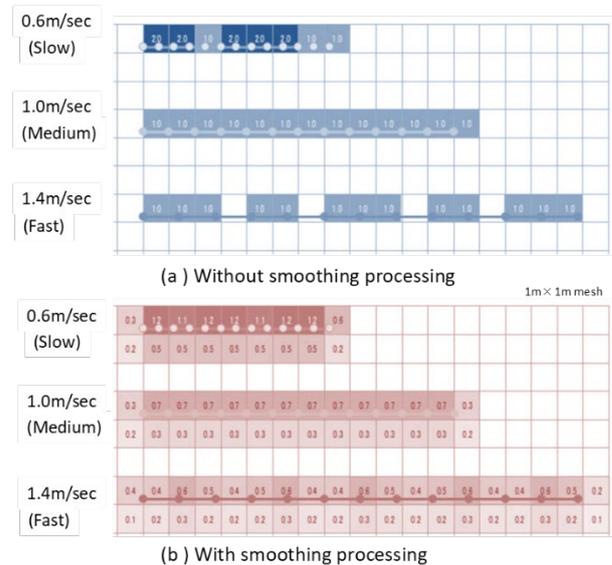


Figure 5. Comparison of data distribution by walking speed with and without smoothing processing

- (2) Trajectory data are divided at the trajectory midpoint into the front layer and the back layer (Fig.4-c) to represent the orientation, that is, where the person came from and where he/she is going.
- (3) In order to represent the location where a person stopped, the data from at least 1 s after smoothing was stopped is copied to the staying layer. If the stop lasts longer than 10 s, time values are normalized to 1.0. In other words,
 - Front layer: 0 s to 1.0 s → 0 to 1.0;
 - Back layer: 0 s to 1.0 s → 0 to 1.0;
 - Staying layer: 1.0 s to 10 s → 0 to 1.0; ≥10 s → 1.0.
- (4) In order to represent the action of sitting down, the initial and final time points of each trajectory are set to 1 at both the time of sitting down and the time of leaving the seat, and all other time points are assigned the value of 0 at the seat layer.

After the above process, the data sizes were as follows: 267 front layers, 274 back layers, 292 staying layers, and 2 seat layers, for a total input layer data length of 835 (Fig. 4-d).

2.3 Issues in setting number of hidden neurons

The number of hidden layers was set to 2, and the method of classifying trajectories by the patterns appearing in the second hidden layer was examined.

(1) Number of neurons in the first hidden layer

The method by which the number of hidden neurons is selected is important. Since Hinton, the originator of the deep belief network model, did not clearly state any way to select this (Hinton.G.E et al., 2006), it is up to the user to calibrate the method to the specific problem at hand. In this study, the number of neurons in the first layer was evaluated and specified on the basis of the following two standards:

① Are the input values sufficiently consistent with the associated values?

The discrepancies between the input and associated values and their correlations were examined to assess whether there are sufficient neurons.

② Is there sufficient information entropy in the hidden layers?

When the probability P_i (the mean value for each trajectory) of ignition of all the hidden neurons after learning is used, the information entropy H is given as follow

$$H = \sum_i (P_i \log_2(P_i) + (1-P_i) \log_2(1-P_i)) \quad \text{Eq. (1)}$$

Using equation (1), it can be verified that the information entropy is efficiently increased by increasing the number of neurons.

(2) Number of neurons in second hidden layer

The way by which things are determined in the first hidden layer also determines the extent of inheritance of the features of the input values, and the way by which things are determined in the second hidden layer determines the classification sizes. If the number of classes expected by the analyst is N_c , then the number of neurons in the second layer could be set by either of the following:

Proposal 1: Number satisfying ($2^{\text{number of neurons}} > N_c$)

Proposal 2: Number satisfying (information entropy of classification result $> \log_2(N_c)$)

Proposal 1 is effective for ideal cases in which there are no biases in the number of data in each class. However, the classes under consideration here show scatter. Therefore, Proposal 2 was selected as the method for determining the number of neurons in the second hidden layer.

3 Evaluation Experiment and Results of Analysis

3.1 Investigation of neuron number on hidden layers

(1) Number of neurons in first hidden layer

The number of hidden neurons was changed from 20 to 160 and it was calculated whether the input values were sufficiently consistent with associated values. Fig.6-a, b shows the number of hidden neurons at intervals of 10,000 and 30,000 learning cycles (epochs) and the final errors and correlation coefficients. The reader can see that the error dropped dramatically once the number of hidden neurons exceeded 40, and that the accuracy fell every time the number of parameters was increased after 10,000 learning cycles. There was also a tendency to obtain greater stability of results when using 30,000 learning cycles.

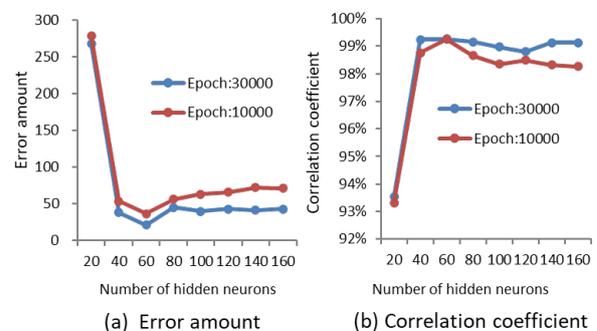


Figure 6. Relation between number of hidden neurons and error-related indicators by Epoch.

Next, the number of hidden neurons was varied while observing how the information entropy in the hidden layers changed. Fig.7-a, b shows how the information entropy in the hidden layers and the information entropy per neuron varied with the numbers of hidden neurons. When that number was 160, the information entropy reached its maximum value (approximately matching information entropy in the input layer, and then stabilizing at that

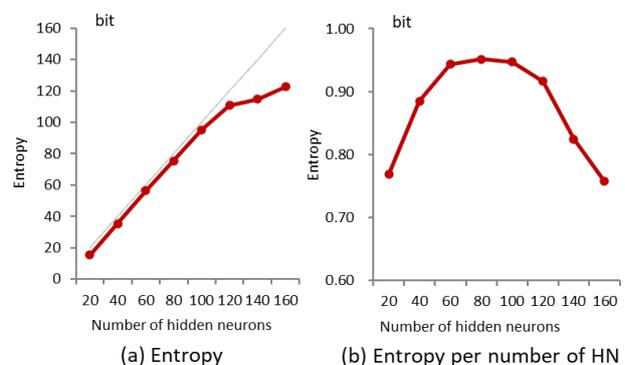


Figure 7. Relation between number of hidden neuron and entropy

value). However, examination of the information entropy per neuron shows that this parameter stabilized at a high level in the 60 to 100 range but fell sharply at higher neuron numbers. As long as the information in the input layer is efficiently propagated, models with lower numbers of parameters stabilize. Thus, lower numbers of neurons are preferable. In this study, therefore, 100 neurons were placed in the first hidden layer, the number just before the sharp decrease in parameter efficiency.

(2) Number of neurons in second hidden layer

The number of neurons on this layer affects the size of the classifications to be calculated. Here, the number of classifications was set at 13, using the results of the previous visual assessment. Since $\log_2(13)$ is approximately 3.7, the number of neurons was initialized at 4 and increased from there. The information entropy of the classification results exceeded $\log_2(13)$ at 8 neurons, so this number was employed for the second hidden layer (Table 1).

Table 1. Number of 2nd Hidden Neurons

model	100 × 4	100 × 5	100 × 6	100 × 7	100 × 8	100 × 9	Log(13)
Entropy	2.99	3.40	3.54	3.67	3.80	4.20	3.70

3.2 Outline of results of machine classification

Trajectory data was learned using 100 neurons in the first hidden layer and 8 neurons in the second hidden layer (100×8 model). Analysis of this model indicated a correlation of 0.990 between the input values and associated values in the first hidden layer, and the correlation was 0.740 in the second hidden layer. Since the second hidden layer employed 8-byte data, theoretically, $2^8 = 256$ classifications were possible, but using RBM classification (machine classification), the trajectories were divided into 25 classifications.

Fig.8-a provides representative diagrams of the classified trajectories (combining different actual individual trajectories that had been classed together).

3.3 Comparison of machine and visual classification, observations

Details of the machine and the visual classifications are examined for comparison here to confirm whether machine learning is a reliable tool capable of providing results similar to the intuition of a human analyst.

Fig.8-b presents a matrix for comparing between the classifications by the machine 100×8 model and the visual classification.

The mean information entropy (data entropy) of the classification results with respect to each axis is calculated as an index for the consistency of the data in order to make

a qualitative assessment of agreement between the vertical axis (machine classification) and the horizontal axis (visual classification). The entropy ratio is then calculated as the fraction of the maximum information entropy (base entropy), which is determined by the number of classifications. A lower value indicates a greater agreement between the 100×8 model and the visual classification. Examination of the results in Fig. 9 shows that the entropy ratio in the visual classification was 0.6%, whereas that in the machine classification was 10.6%. The reader can see that the agreement in this model was superior to that of models with modified numbers of second hidden neurons, i.e., better than the 100×7 or 100×9 model.

Thus, the entropy ratio of the horizontal axis (visual classification) was sufficiently low. Aside from one case, X25, the machine classification matched the several boundaries defined by the visual classification. The machine classification can be considered extremely close to that provided by an intuition of a human analyst. On the other hand, however, the entropy ratio on the vertical axis (machine classification) had a rather high value. The RBM-defined classes tended to have low volumes of data; this was because the RBM tended to distinguish multiple classes within a single visual classification class. Each of the machine-generated classes, X1 through X25, was compared against the visual classifications in order to examine this in more detail.

The machine-generated classes X1 to X4 correspond one-to-one with the visually interpreted classes A1 to A4, which were designated for individuals who travelled directly between the entrance (upper right direction, Up-r) and the patient wing (Down). Every classification contained numerous data sets. Trajectory orientations (Up-r ↔ Down) and splits in individual paths to identical locations (for example, bearing to the right when walking around the escalator) were separated into different classes. It was also possible to distinguish individuals who walked faster than the mean speed, which occurred in every class.

The machine-generated classes X5 to X8 correspond to the visually interpreted class A5, defined as individuals who entered through the entrance, passed the return visit reception machine without stopping, and proceeded to the examination area (Left), and X9 and X10 correspond to A6, who entered through the entrance and stopped at the return visit reception machine before proceeding to the examination area. Although these trajectories are similar at first glance, they are classified according to whether the person stopped at the machine. For example, the “passed without stopping” visually interpreted class A5 corresponds mostly to the machine-generated class X5 (4 cases), but in addition, is also divided into three other classes (X6–X8) with low data counts. The reasons for this were the behaviors of pausing at Reception and again near the seats, entering through the main entrance but going to

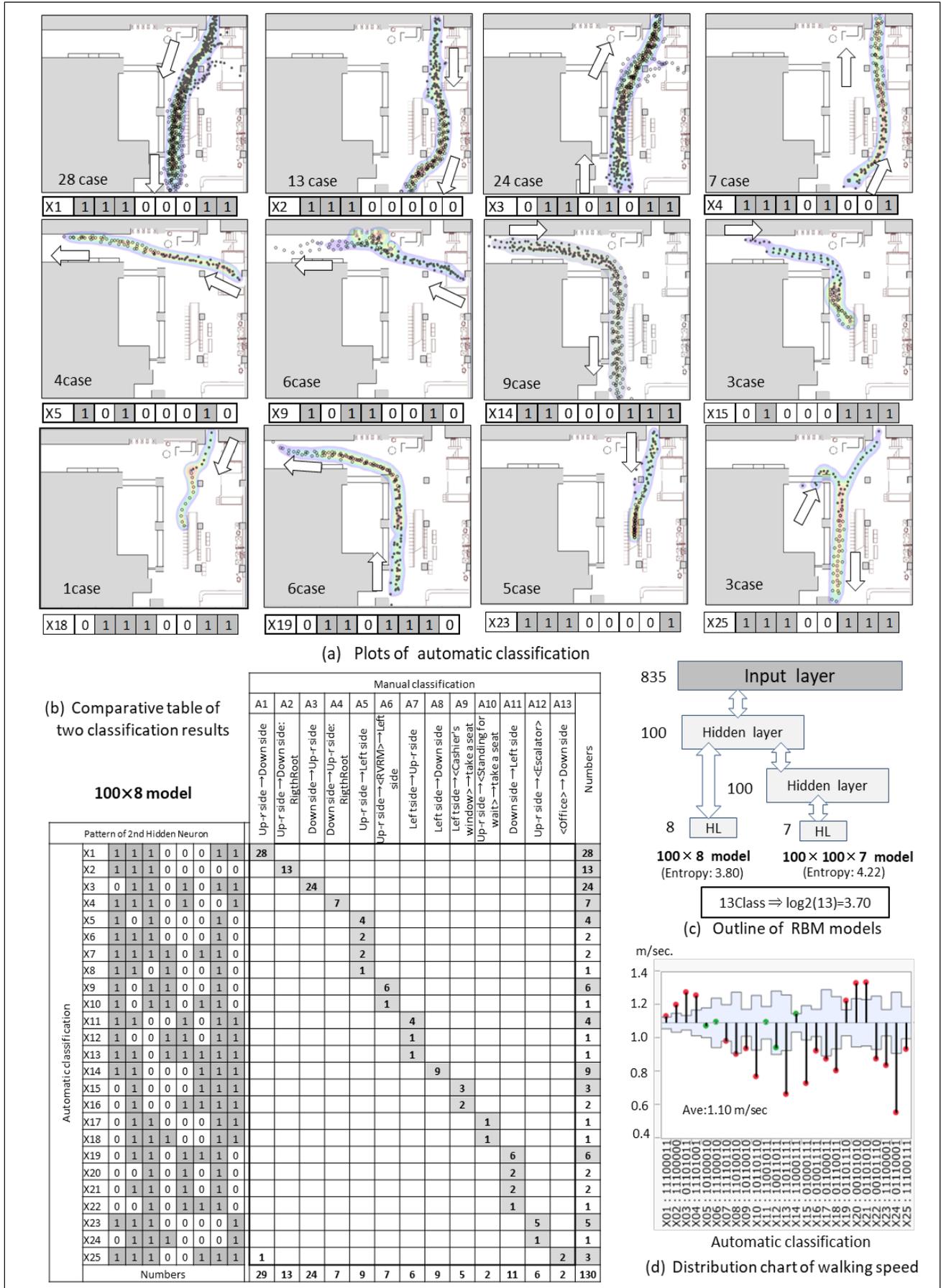


Figure 8. Result of analysis by 100x8 mode

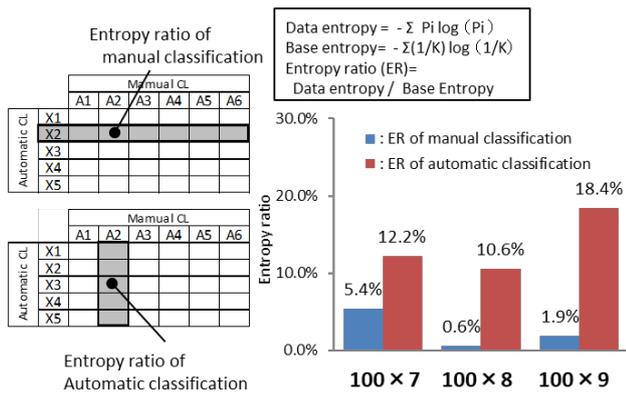


Figure9. Comparison of entropy ratio

from others. These behaviors were not considered by the authors during the visual classification but were carefully distinguished during the machine classification process.

In the same way, the “stopped at the machine” visually interpreted class A6 corresponds mostly to the machine-generated class X9 (6 cases), but class X10 (1 case) is distinguished, due to a different walking speed. These classifications on the vertical axis show a high degree of consistency with the visual classifications from the viewpoints of both orientation and stopping locations. However, differences in walking speeds and other factors that were ignored by the analyst are considered in the classes on the vertical axis. Since these classes were identified as additional categories during machine learning, these reclassification reasons can always be examined afterward with reference to the diagrams of the trajectories, and the classes can be recombined as the analyst deems appropriate.

The machine-generated classes X11 to X13 correspond to the visually interpreted class A7, defined as individuals who entered from the examination area and travelled toward the main entrance (Left side to Up-r), class X14 corresponds to class A8, who entered from the examination area and travelled toward the patient wing (Down), and classes X15 and X16 correspond to class A9, who entered from the examination area, stopped at the cashier’s reception desk, and then, after some time, sat in the adjacent seating area. Patients returning from the examination area show some differences in behavioral patterns from medical staff heading toward the patient wing. The reasons for the splitting seen in these classes were behaviours such as searching for something and hesitating after entering the building, slow walking speed, and differences in lengths of waiting time at the cashier’s reception desk, which allowed differing interpretations.

Classes X17 and X18 correspond to the visually interpreted class A10, defined as individuals who came from a patient wing (Up-r), stopped for some reason, and then sat in the waiting area seats, classes X19 to X22 correspond to class

A11, who came from a patient wing (Down) and went to the examination area, classes X23 and X24 correspond to class A12, who came from a patient wing and went to the escalator, and class X25 corresponds to class A13, who came from an office and proceeded to a patient wing (medical staff). This assignment of features by machine classification was found to strongly reflect the orientation and stopping locations. Even though any given visually interpreted class was divided into multiple classes by the RBM, the reader can anticipate that the analyst could recombine them.

Fig 8-c. shows the mean walking speeds of the various classes identified by the RBM. The statistical test (level of significance $\alpha=0.1$) on the overall mean speed indicated upper and lower limits. There also existed classes that could be called significantly “fast” and “slow”, confirming that speed information was justifiable as a feature for classification.

3.4 Visualization of characteristic patterns of the hidden layers

What features actually were learned in the second hidden layer? In order to determine this, the first hidden layer was inversely estimated using the learned weighting data from the binary values in the second hidden layer. Then, the values in the input layer were inversely estimated. Fig.10 is a visualization of the distributions of typical classes generated by the RBM on the inverse-estimated input layer. The characteristic patterns of orientation and stopping locations are clearly expressed on the front layer, back layer, and staying layer, and the interpretations of those combinations match well with the trajectories in Fig. 8-a. The reader can see that the features latent in the input data were automatically extracted, without any teaching of the trajectory patterns from an exterior agent, and hidden neurons reacting to these patterns were generated.

3.5 Results of classification using three hidden layers

A model containing three hidden layers was created in order to find out what kind of learning takes place. The number of hidden neurons was set at 100 on both the first and second layers, and since the data entropy of the classifications of the third hidden layer was desired, a 100×100×7 model was created, using 7 elements on the third hidden layer, as 7 is the integer higher than $\log_2(13)$ (Table 2). Learning was conducted with the model and the results were compared with those from the 100×8 model.

Table.2 Number of 3rd Hidden Neurons

model	100 × 100 × 5	100 × 100 × 6	100 × 100 × 7	100 × 100 × 8	Log(13)
Entropy	3.33	3.64	4.22	4.28	3.70

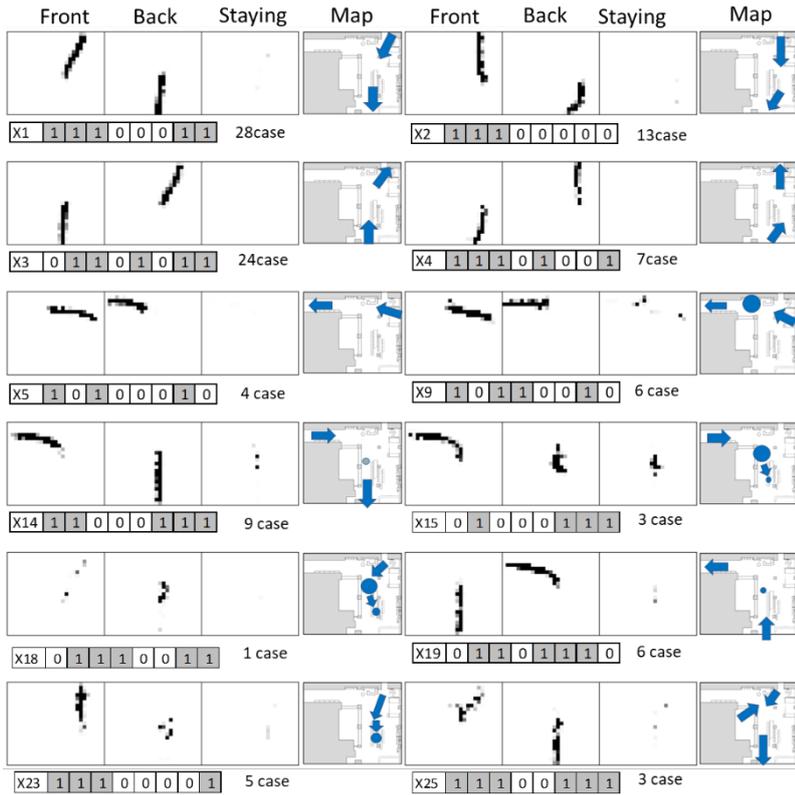
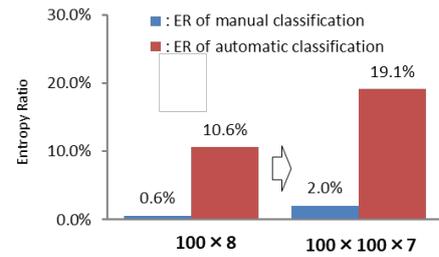


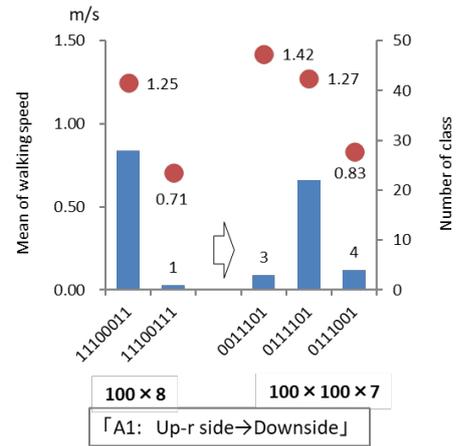
Figure 10. Visualization of feature pattern

Comparing the two entropy ratios, both ratios under visual classification are in the 1% to 2% range, closely resembling each other. However, the entropy ratio of the RBM-classification results ranged approximately 11% to 19% (Fig. 11-a), and the number of classes had grown from 25 to 33. In other words, the number of extracted features increased with the hidden layers, and the bases for classification became more detailed. Examining visually interpreted class A1 (entering from the top right and moving down), for example, the 100×8 model was biased with 27 cases vs. 1 case, while the $100 \times 100 \times 7$ model reclassified it into three classes containing 21, 3, and 4 cases, which indicates a small broadening of the distribution. The difference appeared in the mean walking speeds. In the 100×8 model, the majority of people walked slightly faster than mean speed (1.25 m/s), and there was one outlier who walked slowly (0.71 m/s). In the $100 \times 100 \times 7$ model, three people walked fast (1.42 m/s) and four people walked slowly (0.83 m/s), i.e., the walking speed classification became more finely divided (Fig. 11-b).

Thus, increasing the number of hidden layers increases, in turn, the number of features that can be extracted and used to lend precision to the objective of classification. From the point of view of whether machine learning can function as a device for forming rough classifications similar to those obtained by a human analyst, however, two hidden layers appear to be sufficient for this purpose.



(a) Entropy Ratio by models



(b) Mean of walking speed and number of Class by models

Figure 11. Result of the case of three hidden layers

4. Conclusion

While monitoring behavior with laser sensors is a convenient way to collect large amounts of data about the trajectories of people while walking, it is extremely difficult to classify these data into behavioral patterns by extracting common features from each trajectory. In this study, the classification of the features of pedestrians in the outpatient waiting area of a hospital was selected as an example. The input data indicating orientation and stopping locations are stored in three meshes and were applied in an RBM, which is machine learning tool. The latent features were automatically extracted and a method for classifying pedestrian trajectories was proposed.

A comparison of those results with the visual classifications confirmed that this method is sufficiently functional for classifying foot-travelers in terms of orientation, stopping locations, and walking speed, and provides rough classifications resembling the intuitive judgement of a human analyst. These results also indicate that facility planning can make use of multi-agent simulations to validate designs from the viewpoint of pedestrian data (origin-destination traffic volume, behavioral pattern fractions, and temporal distributions of events).

Data and Software Availability

The hospital survey was conducted under the condition that observation data was not made public, so data cannot be shared.

The software developed in this study cannot be disclosed due to the limitations of Kajima's Intellectual Property Management Regulations.

Acknowledgments

The authors wish to express his sincere thanks for valuable comments from the reviewer of AGILE 2022. This paper is written based on part of Kaneko H and Osaragi T (2017).

References

H.Kaneko, T.Osaragi (2015): Extraction of the spatio-temporal activity of office-workers using laser-scanner trajectory data, *Journal of Environmental Engineering (Transactions of AIJ)* 80(712):559-566 (in Japanese)

H.Kaneko, T.Osaragi (2017): Pedestrian trajectory classification Method by machine learning using data of laser scanner tracking system, *Journal of Environmental Engineering (Transactions of AIJ)* 82(742):1051-1059 (in Japanese)

Vasileia Papathanasopoulou et.al (2021).: Classification of pedestrian behavior using real trajectory data, 7th International Conference on Models and Technologies for Intelligent Transportation Systems, IEEE, DOI: 10.1109 /MT-ITS49943.2021.9529266,2021

Yuko.T, Kuniaki.U (2015): Estimating Semantic Information from Trajectory Data by Unsupervised Learning, The 29th Annual Conference of the Japanese Society for Artificial Intelligence, 2015 (in Japanese)

Junyi Cheng et.al.: An unsupervised approach for semantic place annotation of trajectories based on the prior probability, *Machine Learning (cs.LG); Information Retrieval*

P.Senin: Dynamic Time Warping Algorithm Review, Information and Computer Science Department, University of Hawaii at Manoa, 2008.

Ukai,M.and Nagamine,N.: Development of image Processing Method for Detection of Suspicious Acts, RIRI REPORT, VOL21, No11, pp. 17-22, 2007.11 (in Japanese)

Hinton.G.E,Osindero.S and Teh.Y: A fast learning algorithm for deep belief nets, *Neural Computation*, 18, pp. 1527-1554, 2006

C.M.Bishop:Pattern Recognition and Machine Learning, Springer, 2006

Osawa,M. and Hagiwara,M.:Analysis of Learning Characteristics of RBM and Automatic Method for Deciding the Number of Hidden Neurons, IEICE Technical Report NC2014-22 (2014.10) (in Japanese)