# The Research of Event Detection and Characterization Technology of Ticket Gate in the Urban Rapid Rail Transit

## Yunfeng Hou, Chaoli Wang*, Yunfeng Ji

Department of Control Science and Engineering, University of Shanghai for Science and Technology, Shanghai, China
Email: houyunfeng0930@163.com, *clclwang@126.com, jyf123456789@126.com

## Abstract

**Making events recognition more reliable under complex environment is one of the most important challenges for the intelligent recognition system to the ticket gate in the urban rapid rail transit. The motion objects passing through the ticket gate could be described as a series of moving sequences got by sensors that located in the walkway side of the ticket gate. This paper presents a robust method to detect some classes of events of ticket gate in the urban rapid rail transit. Diffused reflectance infrared sensors are used to collect signals. In this paper, the motion objects are here referred to passenger(s) or (and) luggage(s), for which are of frequent occurrences in the ticket gate of the urban railway traffic. Specifically, this paper makes two main contributions: 1) The proposed recognition method could be used to identify several events, including the event of one person passing through the ticket gate, the event of two consecutive passengers passing through the ticket gate without a big gap between them, and the event of a passenger walking through the ticket gate pulling a suitcase; 2) The moving time sequence matrix is transformed into a one-dimensional vector as the feature descriptor. Deep learning (DL), back propagation neural network (BP), and support vector machine (SVM) are applied to recognize the events respectively. BP has been proved to have a higher recognition rate compared to other methods. In order to implement the three algorithms, a data set is built which includes 150 samples of all kinds of events from the practical tests. Experiments show the effectiveness of the proposed methods.**

## Keywords

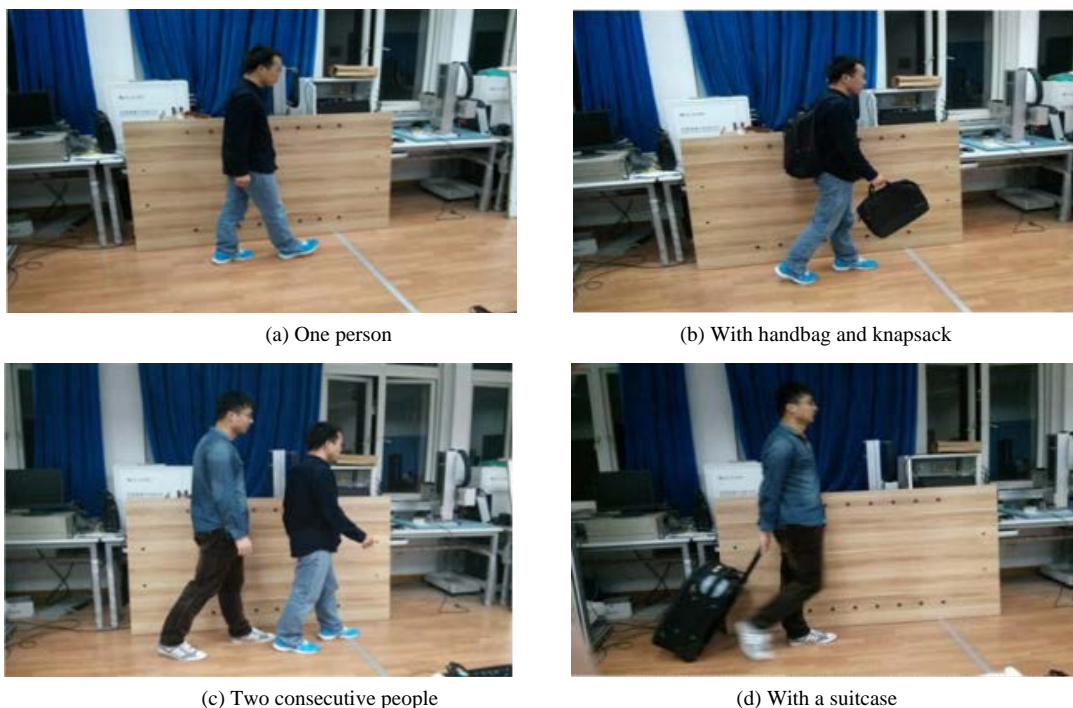**Intelligent Recognition, Ticket Gate, Motion Objects, Infrared Sensors, Time Sequence**
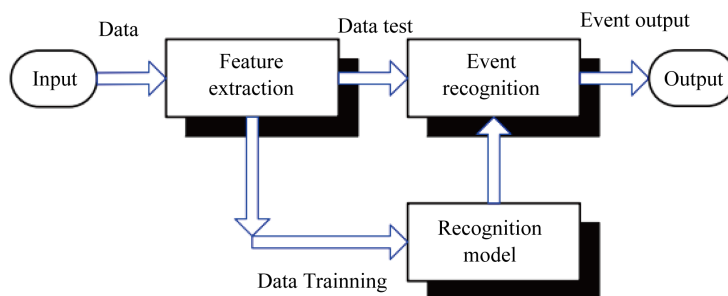
---

*Corresponding author.

## 1. Introduction

The automatic fare collection (AFC) system has received a great deal of attention from the industrial and scientific communities over the past several decades owing to its wide range of applications in preventing stealing a ride, access control, law enforcement, etc. Obviously, it is a significant value study in the intelligent recognition system. Especially in the urban rapid rail transit, the fare evasion phenomenon is coming to rampant. The main difficulty of recognition of these events is that there are many ways for passengers to pass through the walkway of the ticket gate. For example, people can walk through the walkway without anything or pulling a suitcase; two walking passengers also can pass through the walkway closely to each other for a fare evasion. However, as far as I know, few researches have been reported on this problem. Since 1970s, Samsung [1], Gun Nebo [2], Nippon Signal [3], and Omron [4] have successively set up their AFC systems. However, intelligence recognition system was hardly found in their product, or merely equipped with few simple recognition functions. Quri [5] presented a recognition algorithm based on event driven. In that paper, the method combines event recognition technology, human gait recognition technology and human body contour recognition technology. But this method has somewhat difficulty in dealing with the transit without a big gap between two walking passengers. The relatively low recognition rate (85%) is also far from the requirement of market. In order to improve the recognition rate, a novel method is proposed.

This paper focuses mainly on the issue of a robust recognition to several events of ticket gate in the urban rapid rail transit. According to the actual conditions, as shown in **Figure 1**, three types of events are discussed.

The overall process of recognition system can be viewed as a pipeline as shown in **Figure 2**. A single-chip microcomputer is used to collect signals sent by infrared sensors and communicate with an upper computer. During motion objects passing through the walkway, the measurement data from infrared sensors would be combined into a vector according to time order, as shown in **Figure 5**. The vector as the feature descriptor makes the characteristics of events more manifest. All feature descriptors of different events are used to train the



(a) One person

(b) With handbag and knapsack

(c) Two consecutive people

(d) With a suitcase

**Figure 1.** Events defined in this paper: (a) One person without anything walking through the ticket gate; (b) One person with handbag and knapsack walking through the ticket gate; (c) Two consecutive passengers passing through the ticket gate without a big gap between them; (d) A passenger walking through the ticket gate pulling a suitcase. Note that situations described in (a) and (b) are defined into event of one person passing through the ticket gate in this paper; situation described in (c) is defined into event of two consecutive passengers passing through the ticket gate without a big gap between them; situation described in (d) is defined into event of a passenger walking through the ticket gate pulling a suitcase.

**Figure 2.** The overall process of recognition system.

recognition model. Finally, these recognition models are evaluated on the task of events recognition of ticket gate. Various methods [6]-[8] are used to train the recognition model.

Tripod turnstile gate, slide-stile gate and flap-stile gate are the three main types of ticket gates in the market. For slide-stile gate is often used as the preferred ticket gate, in this paper, it will be selected as the main research object. The rest of the paper is organized as follows. Section 2 introduces a topological graph of the sensors-array scheme. Section 3 presents a data preprocessing process. Section 4 is the design of recognition. Section 5 shows experimental results, and Section 6 is a conclusion.

## 2. Topological Graph of Sensors-Array

Eight diffuse reflectance infrared sensors are used to detect the movement sequence of the motion objects in the recognition system. All of the sensors are arranged in an array which we call sensors-array. For every diffuse reflectance infrared sensor, its emitter is integrated with its receiver. When no motion objects detected by a sensor, its output value is 1 (high level signal), otherwise, its output value is 0 (low level signal).

All of the diffuse reflectance infrared sensors are situated in the walkway side of the ticket gate in this paper. As shown in **Figure 3**, the walkway of the ticket gate is divided into three zones, which are defined as import area, gait detection area, export area.

Import area and Export area are used for charging and installing the roller brake respectively, witches are not within the scope of this article.

Gait detection area, No. 1 to No. 8 sensors are 15 cm off the ground, which are used to detect shanks of human body. **Figures 4(a)-(c)** describe the movement sequences of one passenger passing through the walk gate, two consecutive passengers passing through the ticket gate without a big gap between them, and one passenger walking through the walkway pulling a suitcase got by No. 1 to No. 8 sensors respectively, which show a big difference both in data representation and data length. As a whole, the data of above event types show a good sense of distinction.

## 3. Data Preprocessing

In order to improve the recognition rate, it is very important to maintain the integrity of the data. During one passenger passing through the walkway, according to **Figure 4**, the working stage of No. 1 to No. 8 infrared sensors would be represented by a matrix which consists of n rows and 8 columns (every row describes eight sensors' working stage at a point in certain time; every column describes a sensor's working stage at different points in time). Then we translate the matrix into a vector which contains $8 \times n$ elements as the input vector for the event representation, as shown in **Figure 5**. At last, we will unify the input vector to the same length as training vector.

## 4. Event Feature

In this section, machine learning is used to identify events. For a good recognition, the machine learning model must be robust and efficient. Robustness refers to insensitivity to slight difference within the same event. Efficiency refers to a high identification to different events. Although many machine learning models can be used for event identification, the challenge of identifying events is finding a robust and efficient method. In this paper, an extensive set of publicly available machine learning models, deep learning (DL), back propagation neural
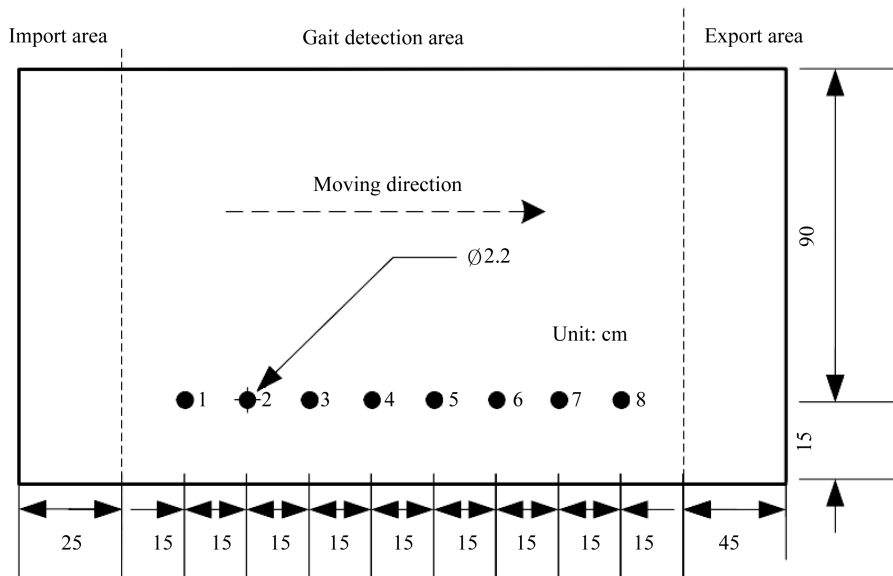
**Figure 3.** Location of diffuse reflectance infrared sensors.



(a) One person (b) Two consecutive persons (c) One person with a suitcase
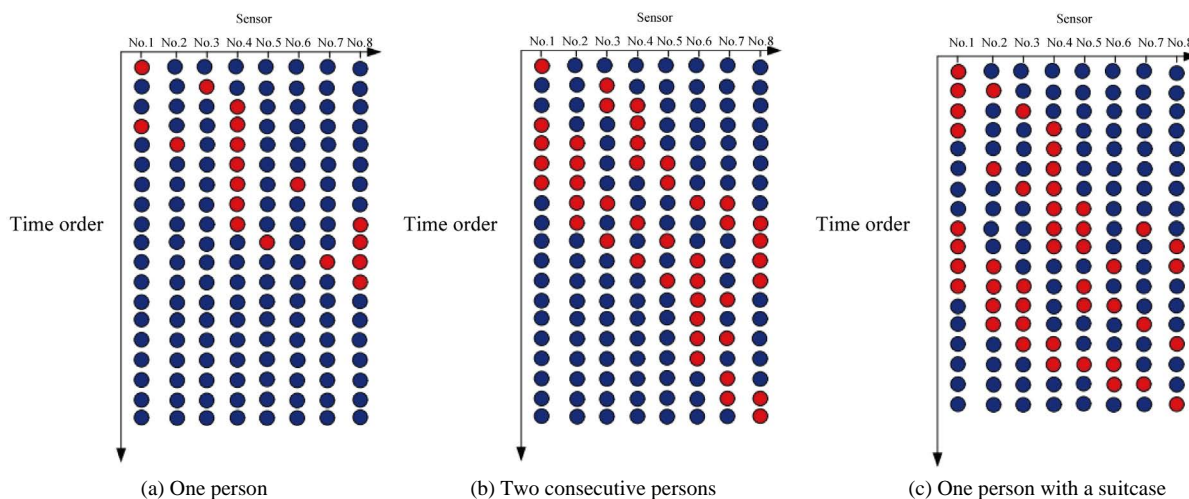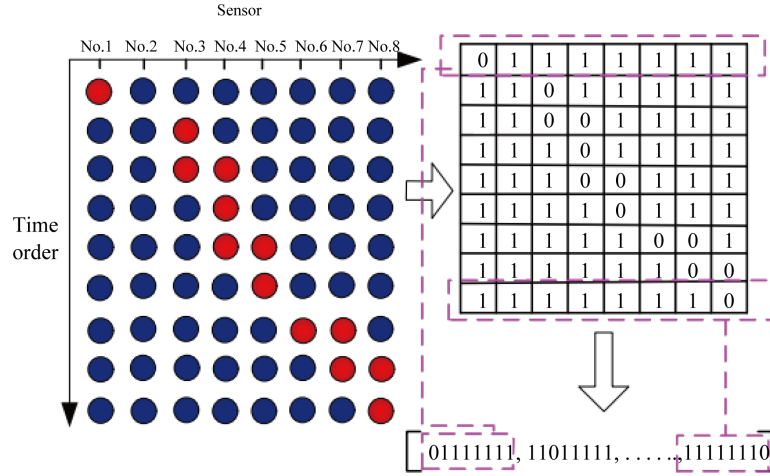
**Figure 4.** Movement sequences of different events defined in this paper got by No. 1 to No. 8 sensors. Red color means sensor detects a motion object, blue color means sensor detects nothing. (a) Movement sequences of one passenger passing through the walk gate got by No. 1 to No. 8 sensors; (b) Movement sequences of two consecutive walking passengers passing through the walk gate without a big gap between them got by No. 1 to No. 8 sensors; (c) Movement sequences of a passenger walking through the walkway pulling a suitcase got by No. 1 to No. 8 sensors.

network (BP), support vector machine (SVM), are used to evaluate the effectiveness of solving recognition problem respectively. Feature descriptor described in the previous section is put into the learning models as the input value.

## 4.1. Deep Learning (DL)

Recently, deep learning (DL) has been the leading technique to learn good information representation that exhibits similar characteristics to that of the mammal brain. It has gained significant interest for building hierarchical representations from unlabeled data. A deep architecture consists of feature detector units arranged in multiple layers: lower layers detect simple features and feed into higher layers, which in turn detect more complex features. In particular, deep belief network (DBN), the most popular approach of deep learning, is a multilayer generative model in which each layer encodes statistical dependencies among the units in the layer below, and it

**Figure 5.** Process of data preprocessing.

can be trained to maximize (approximately) the likelihood of its training data. A great deal of deep belief network (DBN) models have been proposed so far [6] [9]-[11]. Particularly, the model proposed by Hinton *et al.* is a breakthrough for training deep networks. It can be viewed as a composition of simple learning modules, each of which is a Restricted Boltzmann Machine (RBM) that contains a layer of visible units representing observable data and a layer of hidden units learned to represent features that capture higher-order correlations in the data. Today, including machine transliteration [12], hand-written character recognition [9] [13], object recognition [14], various visual data analysis tasks [15]-[17] etc., DL have been successfully applied to above applications.

Restricted Boltzmann Machine (RBM) [13] [18]-[20] is a bipartite, two-layer, undirected graphical model consisting of a set of (binary or real-valued) visible units (random variables) $\mathbf{v}$ of dimension $D$ representing observable data, and a set of binary hidden units (random variables) $\mathbf{h}$ of dimension $K$ learned to represent features that capture higher-order correlations in the observable data. Hidden units $h_j \in \mathbf{h}$ are independent of each other when conditioning on $\mathbf{v}$ since there are no direct connections between hidden units. Similarly, the visible units $v_i \in \mathbf{v}$ are also independent of each other when conditioning on $\mathbf{h}$. **Figure 6** illustrates the undirected graphical model of an RBM.

RBM is an energy-based model. We define the energy of joint distribution over the visible and hidden units by $\mathrm{E}(\mathbf{v},\mathbf{h})$.

$$\mathrm{E}(\mathbf{v},\mathbf{h}) = -\sum_{i=1}^{D}\sum_{j=1}^{K} v_i w_{ij} h_j - \sum_{j=1}^{K} b_j h_j - \sum_{i=1}^{D} c_i v_i \tag{1}$$

where $w_{ij}$ is the weight between the visible units $v_i$ and hidden units $h_j$, $c_i$ and $b_j$ are respectively visible and hidden units bias.

$$p(\mathbf{v},\mathbf{h}) = \frac{1}{Z} \mathrm{e}^{-\mathrm{E}(\mathbf{v},\mathbf{h})} \tag{2}$$
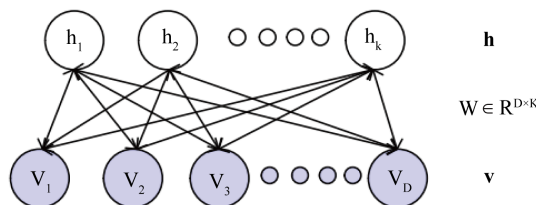
$$Z = \sum_{v,h} \mathrm{e}^{-\mathrm{E}(\mathbf{v},\mathbf{h})} \tag{3}$$

where $Z$ is normalized factor, also known as the partition function.

The probability that the network assigns to a visible vector $\mathbf{v}$ (training vector), is given by summing over all possible hidden vectors:

$$p(\mathbf{v}) = \frac{1}{Z}\sum_{h} \mathrm{e}^{-\mathrm{E}(\mathbf{v},\mathbf{h})}. \tag{4}$$

We can raise the probability that the network assigns to a training vector by adjusting the weights and bias to lower the energy of the training vector. The derivative of the log probability of a training vector with respect to a weight is surprisingly simple.

**Figure 6.** Undirected graphical model of an RBM.

$$\frac{\partial \log p(\mathbf{v})}{\partial w_{ij}} = \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \tag{5}$$

where the angle brackets represent expectations under the distribution specified by the subscript that follows. $\langle \rangle_{\text{data}}$ is the distribution of the training data, and $\langle \rangle_{\text{model}}$ is the equilibrium distribution defined by RBM. We can get a very simple learning rule for performing stochastic steepest ascent in the log probability of the training data:

$$\Delta w_{ij} = \xi \left( \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \right) \tag{6}$$

where $\xi$ denotes a learning rate.

DBN is a probabilistic generative model that composed by several layers of hidden variables, in which each layer captures high-order-correlations between the activities of hidden features in the layer below. DBN is mainly made up by Restricted Boltzmann Machine (RBM) and two adjacent layers of DBN including a RBM, as shown in **Figure 7**; the DBN is composed by three hidden layers $\mathbf{h}^1$, $\mathbf{h}^2$ and $\mathbf{h}^3$, $\mathbf{v}$ is the input layer.

A marked feature of DBN is that to learn a deep hierarchical model we should repeat a greedy layer by layer unsupervised training for several times. In 2006, a fast, unsupervised learning algorithm for these deep networks is proposed by Hinton *et al.* [9].

## 4.2. Back Propagation Neural Network (BP)

BP network is basically a gradient decent algorithm designed to minimize the error function in the weights space. During training of the neural network, weights are adjusted to decrease the total error. In principle, it has been proved that any continuous function can be uniformly approximated by BP network model with one or more hidden layer. However, it is not easy to choose the appropriate number of hidden layers because currently there is no definite rule to determine it. Using too many hidden layers maybe lead to local optimum and increase training time. But using too few hidden layers impairs the neural network and prevents the correctly mapping of inputs and outputs. In the next section, we will test the performance of BP with different numbers of hidden layers.
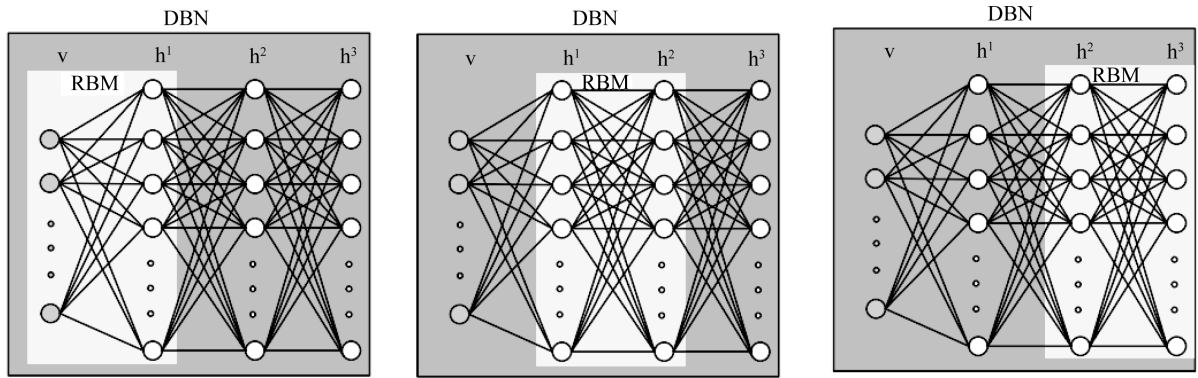
## 4.3. Support Vector Machine (SVM)

Support vector machine (SVM) is a technique developed by Vapnik (1999) [7], which is used to train the classifiers according to structural risk minimizations concept. SVM has been used in various classification problems and pattern recognition field since 90s.

Data can be classified into two classes (positive and negative) with SVM. Assuming a group of points that belong to these two classes, SVM description can be referred as follows: a SVM establishes the hyper plane which allocates the majority of points of the same class in the same side, which maximizes the distance between the two classes to the hyper plane. Define the optimal separating hyper plane as the distance between one class and a hyper plane which is the small distance between the hyper plane and the other points of the same class. The hyper plane created by SVM contains a subset of points of these two classes, which we call support vectors.

Assuming training set $\{\mathbf{x}_i, y_i\}\, i = 1, \cdots, k$, where $\mathbf{x} \in \Re^N$ is an *N*-dimensional input data and $y \in \{-1, +1\}$ is the required classification.

The goal is to estimate a function with the training set that classifies the pairs $\{\mathbf{x}_i, y_i\}$ which haven't been correctly used yet. The optimal hyperplane is defined by:

$$\mathbf{w}x + b = 0 \tag{7}$$

**Figure 7.** Deep belief network with three hidden layers $h^1$, $h^2$ and $h^3$.

where $\mathbf{w} \in R^n$ is the weight vector and $b$ is the bias. A vector $\mathbf{x}_i$ that labeled as the same class $y_i$ must satisfy the follow equation:

$$y_i(\mathbf{wx}_i + b) - 1 \geq 0 \tag{8}$$

where $\mathbf{w}$ is normal to the hyperplane. The Euclidian distance between these points over the separating margin and the hyperplane is determined by Equation (9).

$$\frac{y_i(\mathbf{wx}_i + b)}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|} \tag{9}$$

Thus, for maximizing the separating hyperplane, we should minimize $\|\mathbf{w}\|$. The following equation can be established with Lagrange multipliers theory.

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{N} \alpha_i \left[ y_i(\mathbf{wx}_i + b) - 1 \right] \tag{10}$$

where $\alpha_i$ is the Lagrange multiplier. The optimization problem solution is obtained through minimization of $J(\mathbf{w}, b, \alpha)$.

For nonlinear decision surfaces, the input data generated by machines in a higher dimension space tends to decrease computational efforts of the support machines. By introducing variables that enlarge the margin by relaxing the linear SVM, to allow a few misclassification errors in the margin, and penalize these errors through penalty parameter $C$ in Equation (12), the not linearly separated problem can be solved. These changes allow that the Equation (8) can be violated, or:

$$y_i(\mathbf{wx}_i + b) + \zeta_i \geq 1 \tag{11}$$

where $\zeta_i$ is non-negative variables associated to each training vector. The cost function can be given by Equation (12).

$$\phi(\mathbf{w}, \zeta) = \frac{1}{2}\|\mathbf{w}\| + C \sum_{j=1}^{N} \zeta_j \tag{12}$$

where $C$ is the training parameter that decides the balance between the training error and model complexity, which is known by regulating constant.

A detailed discussion of the computational aspects of SVM can be found in [21] and [22], with many examples also can be found in [23]-[26].

## 5. Experiment

We will illustrate the effectiveness of the method by presenting experiments on three events data got by sensors 1 - 8. The database used in our experiment was constructed by 150 samples (with 50 samples per class). 90 samples (with 30 samples per class) are used to train DL, BP, SVM and 60 samples (with 20 samples per class) are used to test recognition rates.

## 5.1. Experimental Settings

In this paper, deep learning (DL), neural networks based on error back-propagation algorithms (BP), support vector machine (SVM), are used for learning the data respectively. Events of one person with a bag or without anything walking through the ticket gate, two consecutive passengers passing through the ticket gate without a big gap between them, a passenger walking through the ticket gate pulling a suitcase are defined as event 1, event 2, and event 3. The definition of all events refers to the assembly of event 1, event 2 and event 3. The running environment of experiments is Matlab 2013a.
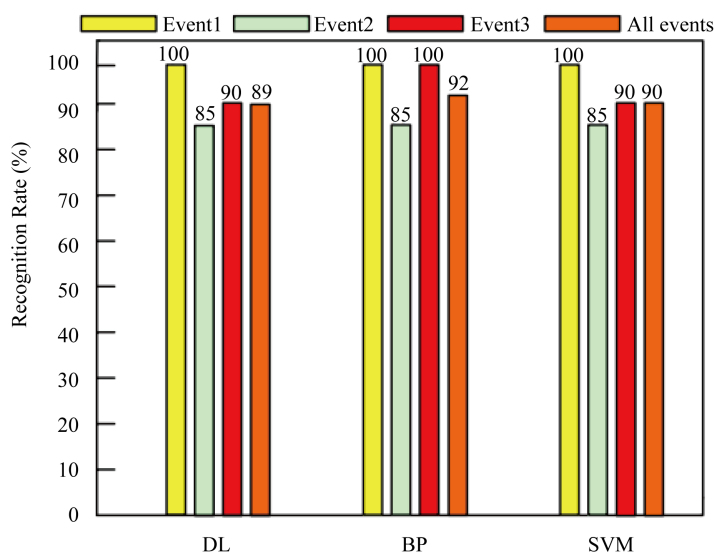
## 5.2. Results for Events Classification

To aid comparison with previous work, **Table 1** summarizes the average recognition rates of the recognition system and Quri's recognition system at a false acceptance rate of 0.1%. Note that the method gives perfect recognition results for database build in this paper: DL gives 89.2%, BP gives 92.5%, and SVM gives 90.0%. While, limited by the fixed sensors' position, method proposed in Quri's paper gets a relatively low recognition rate with 85.0%.

Figure 8 illustrates the recognition performance of recognition manners (DL, BP, SVM) referred in this paper. BP gives better performance than DL, SVM overall. All recognition manners got a 100% recognition rate to event of one person walking through the ticket gate. While all recognition manners got a relatively low recognition rate to event of two consecutive passengers passing through the ticket gate without any gap between them. While, BP shows a higher recognition rates both in events 3 (100%) and all events referred in this paper (92%), with recognition rates of DL and SVM 90%, 89% and 90%, 90%.
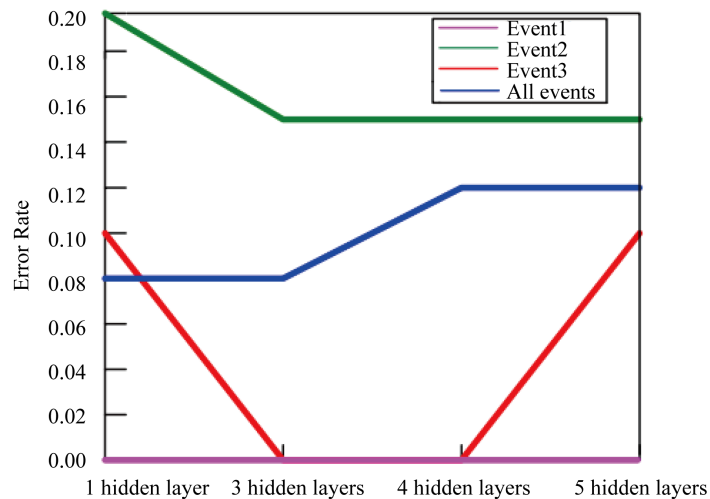
**Table 1.** Average recognition rates obtained with using different recognition methods on database built in this paper. 90 samples (30 samples per class) are used for training and 60 samples (20 samples per class) are used for test. Note that recognition of Quri's methods is got with his recognition scheme and database defined in his paper.

| Method | Average recognition rates | | | |
|---|---|---|---|---|
| | Test samples/train samples | Min | Max | Average |
| Quri [1] | - | - | - | 0.850 |
| DL | 60/90 | 0.833 | 0.900 | 0.892 |
| BP | 60/90 | 0.916 | 0.933 | 0.925 |
| SVM | 60/90 | 0.900 | 0.900 | 0.900 |



**Figure 8.** Recognition performance of methods proposed.

**Figure 9.** Recognition error rates of BP.

In order to have an estimate of the performance of BP with different numbers of hidden layers, we test the recognition rate of BP with 1, 3, 4, 5 hidden layers. The results shown in **Figure 9** are average recognition error rate of BP with 1, 3, 4, 5 hidden layers respectively. The results show that the proposed BP models with 3 hidden layers reach a relatively satisfactory recognition rates. In this paper, the BP model contains three hidden layers by default. In theory, the more hidden layer neurons are often with more high recognition accuracy. But too much of the hidden layer neurons can greatly increase the training time, at the same time, the increase of the hidden layer neurons will cause the loss of the tolerance of the network. So the numbers of neurons in 3 hidden layers are empirically selected as 500, 400, and 100 respectively in this paper.

## 6. Conclusions

This paper refers to the event recognition technology on the ticket gate in the urban railway traffic. We put forward a recognition manner according to the different moving sequence of various events. The main contributions are as follows: 1) We integrate the moving sequence matrix to a one-dimensional vector as the feature descriptor. Several recognition algorithms (DL, BP, and SVM) are applied to recognize the events. It has been found that BP has a higher recognition rate compared to other methods with 3 hidden layers. 2) Several events that often happened in the scene of the ticket gate could be identified effectively in this paper. Experiments proved method proposed in this paper have a higher recognition rate.

Eight sensors are used in the senor-array; the hardware of the intelligent recognition system is very simple and inexpensive. Experiments show that the method gets a satisfactory result.

Moreover, we just provide a new insight into the role of machine learning played in dealing with events recognition of ticket gate in the urban railway station. More events should be defined in the follow-up study. Also, more high recognition accuracy is required in the reality need of engineering. Feature selection or recognition system remains an interesting and important research topic in events recognition.

## Acknowledgements

## References

[1]  http://www.samsungsds.com.cn

[2]  http://www.gunneboentrance.com

[3]  http://www.signal.co.jp

[4] http://www.omron.com

[5] Qu, R., Bao, Y.-R. and Ren, C.-M. (2007) The Study of Event Recognition Technology of Ticket Gate in Urban Railway Traffic. *Sensors and Actuators A*: *Physical*, **134**, 641-649. http://dx.doi.org/10.1016/j.sna.2006.06.053

[6] Hinton, G.E. and Salakhutdinov, R.R. (2006) Reducing the Dimensionality of Data with Neural Networks. *Science*, **313**, 504-507. http://dx.doi.org/10.1126/science.1127647

[7] Vapnik, V. (1999) Support Vector Method for Function Estimation. US Patent 5,950,146. Google Patents.

[8] Hornik, K., Stinchcombe, M. and White, H. (1989) Multilayer Feed forward Networks Are Universal Approximators. *Neural Networks*, **2**, 359-366. http://dx.doi.org/10.1016/0893-6080(89)90020-8

[9] Hinton, G.E., Osindero, S. and Teh, Y. (2006) A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, **18**, 1527-1554. http://dx.doi.org/10.1162/neco.2006.18.7.1527

[10] Hinton, G. (2012) A Practical Guide to Training Restricted Boltzmann Machines. *Momentum*, **9**, 599-619.

[11] Bengio, Y. (2009) Learning Deep Architectures for AI. Foundations and Trends® in Machine Learning.

[12] Deselaers, T., Hasan, S., Bender, O. and Ney, H. (2009) A Deep Learning Approach to Machine Transliteration. *Proceedings of the European Chapter of the Association for Computational Linguistics Workshop on Statistical Machine Translation*, Athens, 30-31 March 2009, 233-241.

[13] Hinton, G.E. (2002) Training Products of Experts by Minimizing Contrastive Divergence. *Neural Computation*, **14**, 1771-1800. http://dx.doi.org/10.1162/089976602760128018

[14] Hinton, G. (2007) To Recognize Shapes, First Learn to Generate Images. In: *Computational Neuroscience*: *Theoretical Insights into Brain Function*, 535-547.

[15] Horster, E. and Lienhart, R. (2008) Deep Networks for Image Retrieval on Large-Scale Databases. *Proceedings of the 16th ACM International Conference on Multimedia*, ACM, Vancouver British Columbia, 643-646.

[16] Chen, E., Yang, X., Zha, H., Zhang, R. and Zhang, W. (2008) Learning Object Classes from Image Thumbnails through Deep Neural Networks. *Proceedings of the International Conference on Acoustics*, *Speech and Signal Processing*, Las Vegas, 31 March-4 April 2008, 829-832.

[17] Liu, Y., Zhou, S. and Chen, Q.C. (2011) Discriminative Deep Belief Networks for Visual Data Classification. *Pattern Recognition*, **44**, 2287-2296. http://dx.doi.org/10.1016/j.patcog.2010.12.012

[18] Fischer, A. and Igel, C. (2012) An Introduction to Restricted Boltzmann Machines. *Progress in Pattern Recognition*, *Image Analysis*, *Computer Vision*, *and Applications*, *Lecture Notes in Computer Science*, **7441**, 14-36. http://dx.doi.org/10.1007/978-3-642-33275-3_2

[19] Smolensky, P. (1986) Information Processing in Dynamical Systems: Foundations of Harmony Theory. In: Rumelhart, D.E. and McClelland, J.L., Eds., *Parallel Distributed Processing*, Vol. 1, MIT Press, Cambridge, 194-281.

[20] Freund, Y. and Haussler, D. (1992) Unsupervised Learning of Distributions on Binary Vectors Using Two Layer Networks. *Proceedings of Advances in Neural Information Processing Systems*, 912-919.

[21] Vapnik, V.N. (1995) The Nature of Statistical Learning Theory. Springer-Verlag, New York. http://dx.doi.org/10.1007/978-1-4757-2440-0

[22] Cristianini, N. and Shawe-Taylor, J. (2000) An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press, Cambridge. http://dx.doi.org/10.1017/CBO9780511801389

[23] Pal, M. and Mather, P.M. (2004) Assessment of the Effectiveness of Support Vector Machines for Hyperspectral Data. *Future Generation Computer Systems*, **20**, 1215-1225. http://dx.doi.org/10.1016/j.future.2003.11.011

[24] Melgani, F. and Bruzzone, L. (2004) Classification of Hyperspectral Remote Sensing Images with Support Vector Machines. *IEEE Transactions on Geoscience and Remote Sensing*, **42**, 1778-1790. http://dx.doi.org/10.1109/TGRS.2004.831865

[25] Foody, G.M. and Mathur, A. (2004) A Relative Evaluation of Multiclass Image Classification by Support Vector Machines. *IEEE Transactions on Geoscience and Remote Sensing*, **42**, 1335-1343. http://dx.doi.org/10.1109/TGRS.2004.827257

[26] Camps-Valls, G. and Bruzzone, L. (2009) Kernel Methods for Remote Sensing Data Analysis. Wiley, Chichester. http://dx.doi.org/10.1002/9780470748992