Neural Network Based Threat Assessment for Automated Visual Surveillance

Tony Jan
Department of Computer Systems
University of Technology, Sydney, Australia
P.O. Box 123, Broadway, NSW, 2007, Australia
E-mail: jant@it.uts.edu.au

Abstract— **In automated visual surveillance systems (AVSS), reliable detection of suspicious human behavior is of great practical importance. Many conventional classifiers have shown to perform inadequately because of unpredictable nature of human behavior. Flexible models such as artificial neural network (ANN) models can perform better; however, computational requirement of ANN models can be prohibitively large for real-time video processing. It is interesting to construct a small-sized ANN classifier that can perform well for threat assessment in video-based surveillance system. In this paper, modified probabilistic neural network (MPNN) is introduced that can achieve reliable classification, with significantly reduced computation. Experiment on visual surveillance application shows that MPNN achieves good classification but with much reduced computation compared to other ANN models. In this application, trajectory profile and motion history image information from the observed human subject are used for threat assessment.**

I. Introduction

In automated visual surveillance systems (AVSS), reliable detection of suspicious or endangering human behavior is of great practical importance. Detection of such human behavior can alert relevant authorities for further investigation.

In recent years, both government and private surveillance industries have installed large number of surveillance cameras over wide areas of interest; however, surveillance monitoring still remains largely human-intensive task. Automation or semi-automation of surveillance video monitoring is important to provide continuous and scalable video surveillance monitoring over wide area [1].

AVSS generally requires reliable combination of image processing and machine intelligence techniques. Image processing techniques are used to provide low level image features, and machine intelligence techniques are used to provide expert decision. Extensive research has been reported on low level image processing techniques such as object detection, recognition, and tracking; however, relatively few research has been reported on reliable classification and understanding of human activities from the video images.

Detection of suspicious human behavior involves modeling and classification of human activities with certain rules [2]. Modeling and classification of human activities are not trivial, because of random and complex nature of human movement. The idea is to partition the observed space of human movements into some discrete states and then classify them appropriately. Apparently partitioning of the observed space is very application-specific, and overall it is hard to predict what will constitutes suspicious or endangering behavior.

One popular approach is to use the state-space based modeling for recognizing particular motion sequences that are *a priori* known to be suspicious [3] [4]. The state-space approach defines each static posture or position as a state and describes a motion sequence by the composition of these states with some transitional probabilities [5]. For activity recognition, the joint probability is calculated through the states (motion sequence), and then the most likely motion sequence is selected for classification.

However, due to unpredictable and random nature of human movement, state-space based models have shown to perform inadequately. Instead, more flexible approaches such as Artificial neural network (ANN) promise to perform better. These stateless data-modeling methods can be trained heuristically to nonlinearly partition the input space. ANNs have shown to perform well in many difficult classification problems.

However, ANNs are often difficult to implement in real-life because of large computation required. Nonetheless, nonlinear partitioning is necessary for many classification of complex systems, and it is reasonable that some approximated nonlinear partitioning could still achieve acceptable classification while reducing computational complexity to a reasonable level. This can be considered an engineering solution where one step back in classification performance would guarantee several steps ahead in overall performance including reduced computational complexity and smaller model size.

In this paper, modified probabilistic neural network (MPNN) [6] which approximates the General Regression Neural Network (GRNN) is introduced to classify the human behavior in a car park as either suspicious or unsuspicious. The GRNN algorithm shows its nearest optimality in Bayesian sense and has proven to work very well for classification. This paper compares the performance of MPNN to other conventional neural network classifiers such as MLP and SOM.

II. Method

In wide-area visual surveillance applications, human "behaviors" are often interpreted from human movement. In this work, the detection of suspicious behavior is performed as follows.

0-7803-8359-1/04/$20.00 ©2004 IEEE

A. Object Detection

First, moving objects in the scene are detected by subtracting a model of the background scene from the current frame. The background model and subtraction are computed here using the approach presented in [7]; however, other background subtraction methods or, alternatively, techniques based on optical flow analysis or frame differencing could be successfully exploited.

B. Object Tracking

The moving objects detected as a result of the background subtraction are typically pedestrians and vehicles, and combinations of them such as groups of people. Since our goal is the classification of pedestrians' behaviors, relevant features from those subjects must be extracted and reliably tracked.

Recent approach has been to use the features from various motion information. In particular for endangering behavior detection, trajectory and gesture profiles of particular human subject were analyzed [8]; and for recognition of violent human acts, motion history images (MHI) was analyzed [9]. Although combining trajectory and MHI profiles is intuitively useful, few research has been reported using these features. In this paper, we use the combination of trajectory-related information and MHI information as the features for AVSS threat assessment system.

C. Behavior Recognition

When different sets of movement information are displayed in the feature space, the constellation of different behaviors form distinct clusters. One or more clusters will correspond to suspicious behaviors and observation of them should be used to generate an output to alert the security person. However, the clustering (partitioning) of these behaviors is a nontrivial task. In the next section, we examine a number of classification models that can be used to partition the feature space.

D. Classifiers

1) Introduction to MPNN: The MPNN was initially introduced by Zaknich et al [6]. It is closely related to Specht's GRNN and his previous work, Probabilistic Neural Network (PNN) [10]. The basic MPNN and GRNN methods have similarities with the method of Moody and Darken [11]; the method of RBF's [12], the CMAC [13]; and a number of other non-parametric kernel-based regression techniques stemming from the work of Nadaraya and Watson [14].

A standard version of the GRNN equation, which is similar to the Nadaraya and Watson equations, is

$$\hat{y}(\underline{x}) = \frac{\sum_{i=1}^{NV} y_i exp\frac{-(\underline{x}-\underline{x}_i)^T(\underline{x}-\underline{x}_i)}{2\sigma^2}}{\sum_{i=1}^{NV} exp\frac{-(\underline{x}-\underline{x}_i)^T(\underline{x}-\underline{x}_i)}{2\sigma^2}} \quad (1)$$

where

$\underline{x}$= input vector

$\underline{x}_i$= single training vector in the input space

σ= single learning or smoothing parameter

y_i= scalar output related to $\underline{x}_i$

NV= total number of training vectors

In above GRNN equation 1, each and every training data pair $\{\underline{x}_i, y_i\}$ is incorporated into its architecture, ($\underline{x}_i$ is a single training vector in the input space, and y_i is the associated desired scalar output). This requires very large computations.

If it can be assumed that there is a corresponding scalar output y_i for each local region of the input space which is represented by a center vector $\underline{c}_i$, then the general algorithm of MPNN given in equation 2 can approximate GRNN equation within acceptable accuracy. The center vectors $\underline{c}_i$ for each cluster can be readily estimated from K-means clustering algorithms [6].

This method reduces the complexity in computation significantly while performing acceptable partitioning for classification. The general algorithm for the MPNN is then:

$$\hat{y}(\underline{x}) = \frac{\sum_{i=0}^{M} Z_i y_i f_i(\underline{x})}{\sum_{i=0}^{M} Z_i f_i(\underline{x})} \quad (2)$$

with

$$f_i(\underline{x}) = exp\frac{-(\underline{x}-c_i)^T(\underline{x}-c_i)}{2\sigma^2} \quad (3)$$

where

$\underline{x}$= input vector

$\underline{c}_i$ = center vector for class i in the input space

y_i= scalar output related to $\underline{x}_i$

Z_i= no. of vectors $\underline{x}_i$associated with each $\underline{c}_i$

M = number of unique centers $\underline{c}_i$

A Gaussian function is often used for $f_i(\underline{x})$ as defined in equation 3. However, many other suitable radial basis functions can be used. Tuning simply involves finding the optimal σ giving the minimum MSE of the network output minus the desired output for a representative tuning set of known sample vector pairs by a convergent optimization algorithm.

2) Introduction to SOM: Kohonen's SOM [15] has been widely used in many applications including financial analysis, data mining, signal processing and image processing. The clustering is performed without any supervised solutions given. The clusters are formed based on self-similarity amongst the given sample data. For more details, refer to [15]. SOM can be a very interesting classifier for automated threat assessment because its evolving nature can adapt to changes in the environment.

3) Introduction to MLP: MLP is a common ANN model which stores its knowledge in its network of parallel weights. These weights can be slowly adjusted through training in such a way as to end up with a set of network weights which will give a satisfactory generalized process model inherent in the training data. The training can be optimized by incrementally adjusting the weights to minimize the Mean Square Error (MSE) of the network outputs compared with desired responses. This can be achieved by using backpropagation-of-error learning which is an optimization technique based on the gradient descent algorithms.

III. EXPERIMENT

A. Overview

In visual surveillance, it is impossible to anticipate all scenarios for testing. In this paper, 800 possible scenarios were simulated as the benchmark to compare the performance of different classifiers.

A number of different scenarios in a car park were tested in this experiment. Each scenario lasts $N_f = 50$ frames at a sample rate S of 5 Hz for $T = 10$ seconds.

Within each scenario, a single person subject enters the car park from the upper left corner and then walks to their car. Once the subject enters the scene, his or her head is located and tracked. The speed of the person's head movement is then calculated and stored for later behavior classification.

Figure 1-D shows a typical example of normal and suspicious user behavior described by their trajectory velocity profile. As is evident, the suspicious users show more erratic walking patterns because they are wandering around the cars or looking into the car windows, with possible intent to steal or damage the cars. However classification of this erratic behavior is not simple, hence the need for some ANN classifiers. Simple thresholding or linear discriminant classifiers have been shown to provide inadequate classification performance because of the random nature of the trajectories [8].

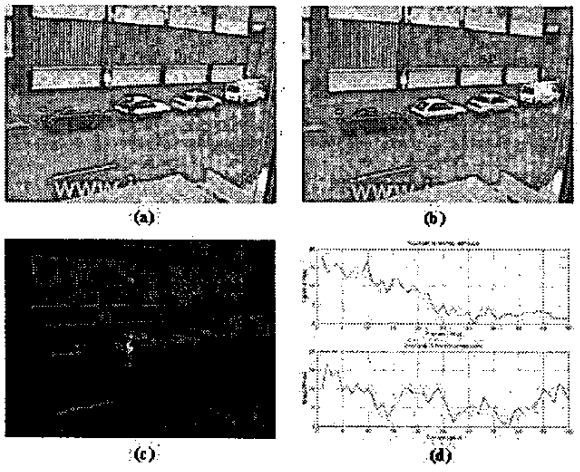

Fig. 1. (a) typical surveillance video image, (b) background image model, (c) detected moving object, and (d) comparison of typical trajectory profiles

It is straightforward to add more classes of behaviors such as the violent or endangering behaviors utilizing additional recognition techniques such as the gesture recognitions or 3D human gesture modelling. In this paper, MHI was calculated for each object in the scenario. The highest, average and median MHI values were profiled for each moving human subject. Threshold value was estimated that separate possible abnormal/violent human movement. This MHI information was fused with the trajectory profile information to assist the decision making in automated threat assessment.

In this paper, 502 scenarios from both normal and suspicious behaviors were used to train the classifiers. For performance comparison, the other 298 independent scenarios were tested.

B. Performance

1) Modified Probabilistic Neural Network: The MPNN had 20 input states and Gaussian Radial Basis Functions as basis function. The speed of the subject's head was quantized to 20 states, and the number of input states was chosen accordingly. The performance of proposed model with VQ-GRNN is reported in Table I.

TABLE I
PERFORMANCE OF MPNN

Input\Classified	Non-suspicious	Suspicious
Non-suspicious behavior	99%	1%
Suspicious behavior	0.25%	99.75%

In applications of this type, the most serious errors occur when a suspicious behavior is misclassified as normal behavior. This corresponds to false dismissals of dangerous situations. Consequently, an optimal classification should reduce this error as much as possible. For the proposed model, this error is limited to less than 1%. This means that over 99% of pedestrians who show suspicious behavior will be detected correctly and the security officer alerted accordingly.

TABLE II
PERFORMANCE OF PROPOSED MODEL IN COMPUTATIONS

Training Time	Testing Time
21 seconds	1.9 seconds

The training and testing times were acceptable for real time implementation in video based surveillance systems. The experiment was performed on Pentium 2 processor with 266 MHz CPU and 256 RAM memory.

2) MultiLayer Perceptron: The MLP network had one input layer with 20 neurons, one output layer with two possible outputs, and one hidden layer with five hidden neurons. The number of neurons in the input layer corresponds to possible head speeds and the other parameters were chosen empirically. The training was performed using the standard back-propagation algorithms. The classification performance is shown in Table III.

TABLE III
PERFORMANCE OF MLP IN CLASSIFICATION

Input\Classified	Non-suspicious	Suspicious
Non-suspicious behavior	98.5%	1.5%
Suspicious behavior	2.5%	97.5%

The accuracy of MLP was very competitive. As well, the time required at run time to perform the classification of a scenario was acceptable.

The only drawback might be its long training time (in the order of a few hundreds of seconds with a normal PC), which

TABLE IV

PERFORMANCE OF MLP IN COMPUTATIONS

Training Time	Testing Time
110 seconds	3 seconds

is expected to grow more than linearly with more training examples. This long training time may affect widespread use in real life applications where the training data set would be considerably larger than in this experiment.

3) Self Organizing Map: The SOM had one input layer with 20 neurons, one output layer with two possible outputs, and one hidden layer with eight learning units. The number of neurons in the input layer corresponds to possible head speeds and other parameters were chosen empirically. The training was performed by the standard distance-based training algorithms. SOM's accuracy was also competitive. The classification performance of SOM is shown in Table V.

TABLE V

PERFORMANCE OF SOM IN CLASSIFICATION

Input\Classified	Non-suspicious	Suspicious
Non-suspicious behavior	95.5%	4.5%
Suspicious behavior	6.25%	93.75%

The training and testing time required for SOM are relatively longer; however, the times are acceptable for real time implementation with offline training. The major difficulty was that shaping of the training data was required to achieve a reasonably simple network architecture. The structure of this classifier depends solely on the sample data provided, hence the network was very sensitive to the available sample data. In addition, many errors could easily be introduced in the initialization stage. The SOM, in this application, did not prove to be adequately reliable.

TABLE VI

PERFORMANCE OF SOM IN COMPUTATIONS

Training Time	Testing Time
412 seconds	9 seconds

C. Results and Analysis

In the previous section, performance of three popular ANN classifiers have been presented in terms of classification accuracy and computational time. The results show that newly proposed MPNN achieves comparable classification performance to MLP and SOM while reducing the computational requirements.

IV. CONCLUSION

This paper has evaluated the use of ANN-based classifiers for assessment of abnormal or suspicious behaviors in an automated visual surveillance application. The experiments shows that the ANN-based classifiers can attain good classification performance, with the recently introduced MPNN achieving significant reduction in computational requirements.

Further research may combine SOM with MPNN in order to make the model size more compact. Also recent popular model such as Support Vector Machine (SVM) may be used to enhance classification performance. Additional features such as gesture and facial expression can improve the classification performance of the automated threat assessment system.

REFERENCES

[1] I. Pavlidis, V. Morellas, P. Tsiamyrtzis, and S. Harp, "Urban surveillance systems: from the laboratory to the commercial world," *Proc. of the IEEE*, vol. 89, pp. 1478–1497, 2001.

[2] E. B. Koller-meier and L. Van Gool, "Modeling and recognition of human actions using a stochastic approach," in *Proc. of 2nd European Workshop on Advanced Video-Based Surveillance Systems*, London, UK, September 2001, pp. 17–28.

[3] A. F. Bobik and A. D. Wilson, "A state-based techniques for the summarization and recognition of gestures," in *Proc. of the 5th International Conference on Computer Vision*, Boston, Massachussetts, USA, 1995, pp. 382–388.

[4] T. Starner and A. Pentland, "Visual recognition of american sign language using hidden markov models," in *Proc. of the Intl. Workshop on Automatic Face and Gesture Recognition*, Zurich, 1995, pp. 189–194.

[5] N. H. Goddard, "Incremental model based discrimination of articulated movements from motion features," in *Proc. of the IEEE Computer Society Workshop on Motion of Non-Rigid and Articulated Objects*, Austin, TX, USA, 1994, pp. 89–95.

[6] A. Zaknich, "Introduction to the modified probabilistic neural network for general signal processing applications," *IEEE Trans. on Signal Processing*, vol. 46, no. 7, pp. 1980–1990, July 1998.

[7] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Statistic and knowledge-based moving object detection in traffic scenes," in *Proc. of the 2000 IEEE Intelligent Transportation Systems Conference*, Dearborn, MI, USA, May 2000, pp. 27–32.

[8] T. Jan, "Neural network classifiers for video surveillance," in *Proc. of IEEE International Workshop on Neural Networks for Signal Processing (IEEE-NNSP)*, Toulouse, France, 2003, pp. 112–121.

[9] J. W. Davis, "Hierarchical motion history images for recognizing human motion," in *Proc. the IEEE Workshop on Detection and Recognition of Events in Video*, Vancouver, Canada, July 2001, pp. 39–51.

[10] D. F. Specht, "Probabilistic neural network," *Int. Journ. Neural Networks*, vol. 3, pp. 109–118, 1990.

[11] J. Moody and C. Darken, "Learning localized receptive fields," in *Proc. of the Connectionist Models Summer Sch.*, Touretzky, Hinton, and Sejnowski, Eds., pp. 133–143. Morgan-Kaufmann, San Franciso, 1988.

[12] D. S. Broomhead and D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks," *Royal Signals Radar Est. Memo*, vol. 4248, Mar. 1988.

[13] J. S. Albus, "A new approach to manipulator control: The cerebellar model articulation controller (cmac)," *J. Dyn. Syst. Meas. Contr.*, pp. 220–227, Sept. 1975.

[14] E. A. Nadaraya, "On estimating regression," *Theo. Probab. Appl.*, vol. 9, pp. 141–142, 1964.

[15] T. Kohonen, *Self-Organising Maps*, Springer-Verlag, 1995.