2011 Sixth International Conference on Image and Graphics

Abnormal Behavior Detection via Sparse Reconstruction Analysis of Trajectory

Ce Li, Zhenjun Han, Qixiang Ye, Jianbin Jiao+
Graduate University of Chinese Academy of Sciences
Beijing, China
+Corresponding Author: Fax: +86-10-88256278, e-mail: jiaojb@gucas.ac.cn

Abstract—**This paper proposes a new method for abnormal behavior detection in surveillance videos via sparse reconstruction analysis. The motion trajectories of objects are firstly defined as fixed-length parametric vectors based on approximating cubic B-spline curves. Then the vectors are classified as behavior patterns and finally distinguished between normal and abnormal behaviors based on sparse reconstruction analysis, in which a classifier is constructed with sparse linear reconstruction coefficients by computing L_1-norm minimization and sparse reconstruction residuals learning from labeled training samples. Experimental results on public dataset show the effectiveness of the proposed approach.**

Keywords-abnormal behavior detection; trajectory representation; sparse reconstruction analysis; L_1-norm minimization

I. INTRODUCTION

In recent years, abnormal behavior detection in surveillance video has become an important issue in computer vision research and vision-based intelligent surveillance application. The analysis of trajectory is a kind of promising method to detect abnormal behavior. Such trajectory-based approaches for behavior analysis consist of two basic elements, motion trajectory representation and behavior patterns classification.

Trajectory representations include representative sequences corresponding to motion vectors [1], motion vectors with acceleration information [2], fixed-length vectors based on re-sampling and linear interpolation [3], and prototype symbolic sequences based on rule-based path segments [4] and [5]. However, the above approaches often result in a redundant trajectory vector. To solve this problem, Naftel [6] proposed function approximation techniques, including Least Square Polynomial, Cheybyshev Polynomial, and DFT (Discrete Fourier Transform), to describe a given trajectory. Recently, a more efficient trajectory representation was proposed by Sillito and Fisher[7] and [8], which adopted Haar Wavelet Coefficients and LCSCA (Least-squares Cubic Spline Curves Approximation) as parametric vectors, for it is easier to model the distribution of normal motion trajectories generally occurring in a particular scene.

The extracted trajectory representations can be used for behavior classification and abnormal detection. In [1], they directly compared new trajectory vectors with a set of clusters. On the basis of this, Self-Organizing Map Neural Network [2], hierarchical Fuzzy K-means clustering [3], hierarchical HMM (Hidden Markov Models) [4] and Nonparametric Bayesian Models [5] are proposed to learn and classify trajectory patterns. Although all of them could detect abnormal trajectory, these approaches need adequate training samples for each trajectory pattern, which brings vast offline labeling work for each surveillance circumstance.

Considering the amount of behavior patterns in a surveillance scene, the most compact reconstruction based on all behavior patterns for a given input trajectory is typically sparse. Inspired by the development of sparse reconstruction in face recognition [9] and object tracking [10], we cast the trajectory classification as finding sparse reconstruction coefficients and residuals based on a constructed trajectory sample set, each of which is represented as a fixed-length vector based on the control points of LCSCA. Combining with L_1–norm minimization, intuition behind the sparse reconstruction lies in the fact that its coefficients imply the discriminative information between trajectory patterns, which are indicated by that some coefficients compactly expressing the input trajectory are nonzero and the other coefficients are almost zero. This provides a new way for abnormal behavior detection in surveillance video. In this paper, we validate the performance of the proposed approach on a real-world dataset. What is more, our approach is still effective when the training sample set is small (even there is only five samples for each pattern) compared with the traditional approaches.

The rest of paper is organized as follows. Section II introduces the LCSCA based trajectory representation and describes the proposed sparse reconstruction analysis for trajectory classification and detection in details. Section III presented the experiments and section IV concludes the paper and discusses future work.

II. METHODOLOGY

In the training procedure, given several motion trajectories of each behavior patterns offline (obtained from an object tracking algorithm [10], or a motion detection method [11], or labeled manually), we constructed the sample set representing the behavior patterns of the surveillance scene and built a classifier based on the discriminative ability of sparse reconstruction analysis,, based on which we classify the behavior patterns in the detection/classification procedure. Given a learned threshold, the classifier can also distinguish between the normal and abnormal behavior patterns. The framework of the proposed approach is shown in Fig. 1.

978-0-7695-4541-7/11 $26.00 © 2011 IEEE
DOI 10.1109/ICIG.2011.104

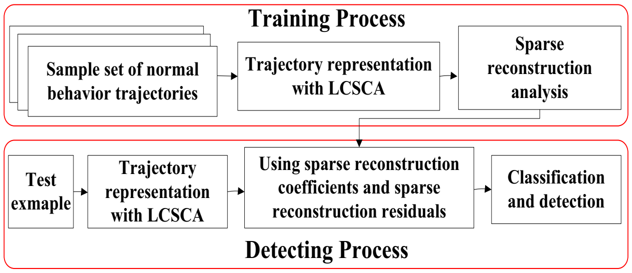

Figure 1. Framework of the proposed approach.

A. Trajectory Representaion

Given a trajectory sequence in (*x*, *y*, *t*) space, we can use the control points of LCSCA as fixed-length parametric vectors to represent the behavior pattern of the objects. The representation is on the basis of the cubic B-spline curve [12], a good feature of which is that its number of control points and weight factors are flexible for any simple or complicated shape of curves. We use B-spline control points to represent both the shape and spatiotemporal profile of a trajectory $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_{t-1},y_{t-1}),(x_t,y_t)\}$ in a parametric way $F=\{C_1^X,C_2^X,\cdots,C_p^X,C_1^Y,C_2^Y,\cdots C_p^Y\}$, illustrated in Fig. 2, where *p* is the number of control points and *t* is the length of trajectory [7].

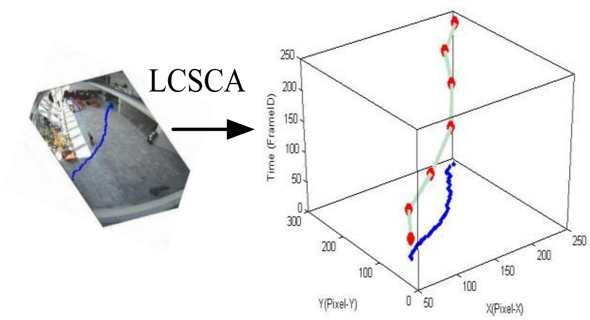

Figure 2. Trajectory representation when p=7.

The transformation procedure of the trajectory representation is as follows:

- Define the parameter vector $s=\{0,s_2,\cdots s_{t-1},s_t\}$, $s_n=\frac{\sum_{i=2}^{n}\sqrt{(x_i-x_{i-1})^2+(y_i-y_{i-1})^2}}{\sum_{i=2}^{t}\sqrt{(x_i-x_{i-1})^2+(y_i-y_{i-1})^2}}(n=2,3,\cdots,t,\quad s_n\in(0,1])$, in which $\sum_{i=2}^{n}\sqrt{(x_i-x_{i-1})^2+(y_i-y_{i-1})^2}$ is the total distance traversed at a given point (x_n,y_n); And define the knot vector:

$$\tau=\left\{\underbrace{0,0,0,0}_{1\cdots4},\underbrace{\frac{1}{p-3},\frac{2}{p-3},\cdots,\frac{p-4}{p-3}}_{5\cdots p},\underbrace{1,1,1,1}_{p+1\cdots p+4}\right\} \quad (1)$$

- Calculate the cubic B-spline basis function with the recursive (2) according to De-Boor algorithm [12]:

$$\begin{aligned} B_{p,1}(s_n)&=\begin{cases}1 & if\,\tau_p\le s_n<\tau_{p+1}\\ 0 & otherwise\end{cases}\\ B_{p,4}(s_n)&=\frac{s_n-\tau_p}{\tau_{p+3}-\tau_p}B_{p,3}(s_n)+\frac{\tau_{p+4}-s_n}{\tau_{p+4}-\tau_{p+1}}B_{p+1,3}(s_n)\end{aligned}. \quad (2)$$

- Find the control points as in (3), which can minimize the sum of squared errors between the original trajectory and its approximation:

$$F^{XY}=\Phi^{\dagger}T^{XY}. \quad (3)$$

where $\Phi=\begin{Bmatrix}B_{1,4}(s_1) & \cdots & B_{p,4}(s_1)\\ \vdots & \ddots & \vdots\\ B_{1,4}(s_t) & \cdots & B_{p,4}(s_t)\end{Bmatrix}$ and $\Phi^{\dagger}=(\Phi^{T}\Phi)^{-1}\Phi^{T}$.

B. Sparse Reconstruction Analysis

A key component of the proposed framework is the sparse reconstruction analysis procedure, in which a test sample is represented with a linear combination of training samples. Supposing there are *J* behavior patterns in a surveillance scene, and each pattern $A_j=\{a_j^1,a_j^2,...,a_j^K\}$ holds *K* training behavior trajectories, we can have the union training set in a particular scene as follows:

$$B=\bigcup\{A_j\}=\{a_1^1,a_1^2,...,a_1^K,a_2^1,a_2^2...a_2^K,...,a_J^K\},\quad j=1...J. \quad (4)$$

For a test sample F_{unknow}, we calculate the sparse linear reconstruction coefficients ψ on set *B* by computing L1-norm minimization as in (5) and (6):

$$B\psi\approx F_{unknow}. \quad (5)$$

$$\arg\min\|\psi\|_1,\ s.t.\ B\psi=F_{unknow} \quad . \quad (6)$$

where $\psi=\{\psi_j^k\},j=1,...,J,k=1,...,K$ and ψ_j^k is the sparse reconstruction coefficient corresponding to the k^{th} sample of j^{th} class in *B*.

To build a classifier, we define a characteristic function δ_j for each class, which keeps the sparse coefficients corresponding to j^{th} class in *B* and sets the coefficients to zero of the other classes. Then the sparse reconstruction residuals based on each trajectory class can be defined as (7). And a threshold θ is used to discriminate the new example as abnormal if the result is negative, as in (8). Finally, if the new example is regarded as a normal trajectory, the trajectory pattern can be classified according to (9). The threshold θ is set as 0.03 empirically in this paper. Fig. 3 shows the sparse reconstruction coefficients, the reconstruction residuals based on each class and proportion of the trajectory belongs to class 7.

$$r_j(F_{unknow})=\|F_{unknow}-B\delta_j(\psi)\|_2,\quad j=1\ldots J. \quad (7)$$

$$\text{Detect}(F_{unknow}) = \text{sign}(\min_{j=1\ldots J} H_j - \theta), \quad where\ H_j = \frac{1/\ r_j(F_{unknow})}{\sum_{i=1}^{J} 1/\ r_j(F_{unknow})}. \quad (8)$$

$$\text{Classify}(F_{unknow}) = \underset{j}{\text{argmin}}\ r_j(\text{F}_{unknow}) \quad . \quad (9)$$

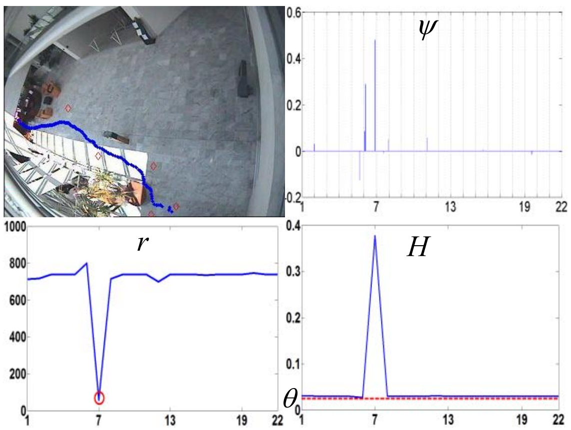

Figure 3. Sparse reconstruction coefficients (top-right), residuals (bottom-left) and proportion of the trajectory belongs to class 7.

III. EXPERIMENTS

In this section, we carried out experiments to validate the proposed method with the publicly available CAVIAR [13] dataset which contains a series of behaviors in the entrance lobby of INRIA lab. There are 11 entry-exit routes appeared in the test data. Considering the traversal directions, we obtained totally 22 types of normal trajectories for behavior analysis (each type holding 100 simulated tracks shared by Rowland Sillito[1]). Examples from each of the routes from dataset are shown in Fig. 4. In the detecting procedure, we selected 21 trajectories to represent the normal behavior, consisting of people walking directly from one exit to another, and 19 trajectories to define abnormal behavior, consisting of people fight, falling down and leaving or collecting packages, as test samples.

For trajectory representation, we defined two kinds of representation sequence: *R1* and *R2*. *R1* is defined as the sequence $F = \{C_1^X, C_2^X, \cdots, C_p^X, C_1^Y, C_2^Y, \cdots C_p^Y\}$ calculated by the method in part A of section II; *R2* contains F and the coordinates of the entry and exit positions. As for *p*, it is directly bound up with the number of control points. We set $p = \{5, 7, 9, 11\}$ for performance comparisons.

To quantitatively validate the proposed approach, *DACC* (Detection ACCuracy) and *CCR* (Correct Classification Rate) are defined as follows:

$$DACC = \frac{TP + TN}{The\ total\ of\ test\ samples}. \quad (9)$$

[1] Thanks to the kindly help of Rowland Sillito about the dataset, which were used for anomalous trajectory detection in [7].

$$CCR = \frac{The\ number\ of\ outcome\ in\ correct\ classification}{The\ total\ of\ normal\ test\ samples}. \quad (10)$$

where *TP* (True Positives) is the number of trajectories correctly detected as normal when it is in fact normal, and *TN* (True Negatives) is the number of trajectories correctly detected as abnormal when it is in fact abnormal.

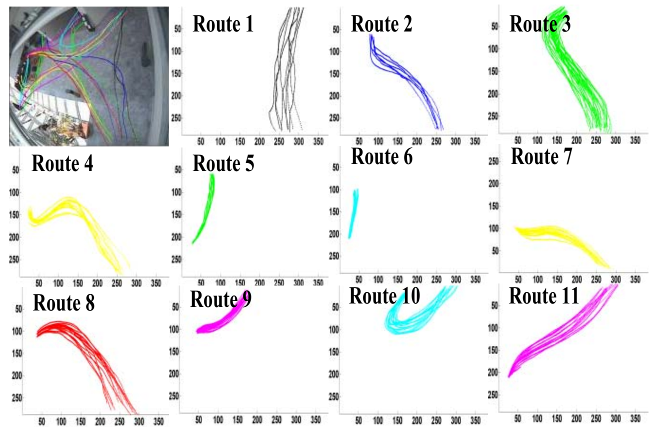

Figure 4. Samples from each route in dataset.

The following experiments based on *R1* and *R2* representations for each trajectory are run on the PC with Intel Core(TM) i7CPU(2.93GHz) and 4GB RAM.

A. Trajectory analysis based on R1

(a)

(b)

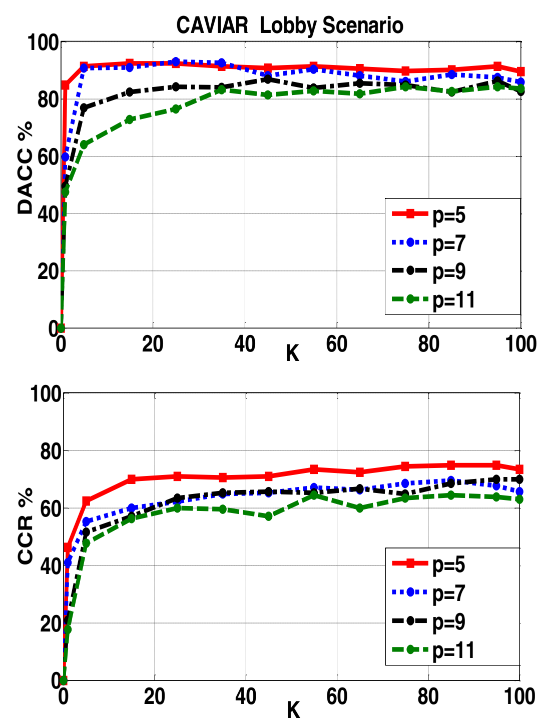

Figure 5. Results of (a) DACC, and (b) CCR based on R1.

The average DACC and CCR of 10 trails based on *R1* are shown in Fig.5, which shows the variations of DACC and CCR as *K* increases, where *K* is the number of the training samples in each class. In the CAVIAR lobby scenario, if we vary the number of control points *p*, it can be seen that when *p* is set as 5, a better DACC=90.42%($\pm$3.85%) and CCR=70.09%($\pm$6.13%) are obtained (the red line in Fig. 5). Even DACC (84.75%) and CCR (62.38%) can be obtained when *K* is set as small as 5.

B. *Trajectory analysis based on R2*

The results when adding the coordinates of the entry and the exit as the R2 representation are shown in Fig. 6. When *p*=5, DACC=83.75% ($\pm$2.88%) and CCR=68.15% ($\pm$4.53%) are still steadily higher than the others. Besides, a good DACC (81.75%) and CCR (63.81%) still can be obtained when *K* =5. Comparing the results shown in Fig. 5, *R2* may too sensitive in some cases, since trajectories sharing the same entry or exit can sometime been seen as the same behavior pattern.

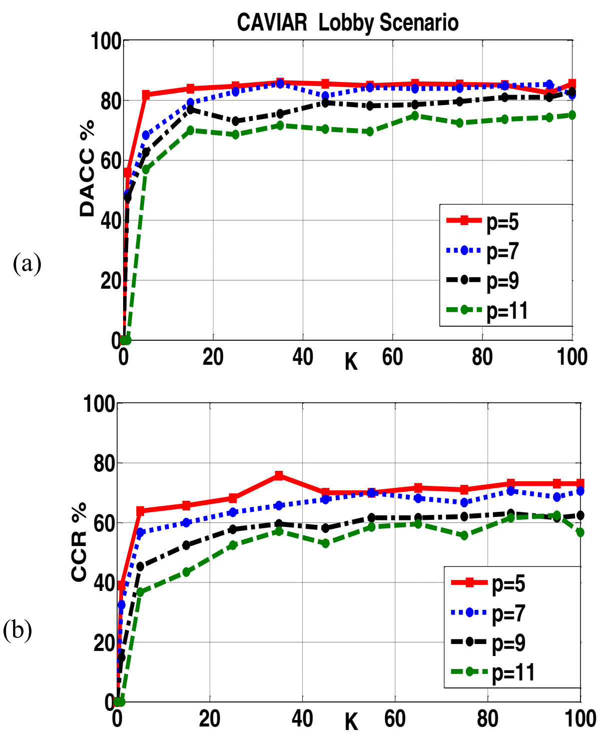

Figure 6. Results of (a) DACC, and (b) CCR based on R2.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a trajectory-based abnormal behavior detection approach. The new concept and technique introduced in this paper is trajectory sparse reconstruction analysis, which focuses on sparse linear reconstruction coefficients and residuals learned from labeled data in a surveillance scene. Experimental results on a real-world dataset have shown that the performance of our method on different sizes of training samples, which is still effective even on small training samples (less than 10).

A known issue in the proposed method is that the performance is sensitive when choosing different number of control points, which should be further considered in the later work.

ACKNOWLEDGMENT

This work is supported by National Basic Research Program of China (973 Program) with Nos. 2011CB706900, 2010CB731800, and National Science Foundation of China with Nos. 60872143, 61039003.

REFERENCES

[1] N. Johnson, and D. Hogg, “Learning the distribution of object trajectories for event recognition”, Image Vision Computing, 1996, pp. 609-615.

[2] J. Owens, and A. Hunter, “Application of the self-organizing map to trajectory classification”, IEEE International Workshop on Visual Surveillance, 2000, pp. 609-615.

[3] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S.J. Maybank, “A system for learning statistical motion patterns”, IEEE Transaction on Pattern Analysis and Machine Intelligence, 2006, pp. 1450-1464.

[4] D. Makris, and T. Ellis, “Learning semantic scene models from observing activity in visual surveillance”, IEEE Transactions on Systems, Man, and Cybernetics – Part B, 2005, pp. 397-408.

[5] X. Wang, and E. Grimson, “Trajectory analysis and semntic region modeling using a nonparametric bayesian model”, IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[6] A. Naftel, and S. Khalid, “Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space”, Multimedia Systems, 2008, pp. 227-238.

[7] R.R. Sillito, and R.B. Fisher, “Semi-supervised learning for anomalous trajectory detection”, in Proceedings of British Machine Vision Conference, 2008, pp. 1035-1044.

[8] R.R. Sillito, and R.B. Fisher, “Parametric trajectory representations for behaviour classification”, in Proceedings of British Machine Vision Conference, 2009.

[9] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Y. Ma, “Robust face recognition via sparse representation”, IEEE Transaction on Pattern Analysis and Machine Intelligence, 2009, pp. 210-227.

[10] Z.J. Han, J.B. Jiao, B.C. Zhang, Q.X. Ye, and J.Z. Liu, “Visual object tracking via sample-based adaptive sparse representation (AdaSR)”, Pattern Recognition, 2011, pp. 2170-2183.

[11] B.C. Zhang, Y.S. Gao, S.Q. Zhao, and B.N. Zhong, “Kernel similarity modeling of texture pattern flow for motion detection in complex background”, IEEE Transactions on Circuits and System for Video Technology, 2011, pp. 29-38.

[12] T. Hastie, R. Tibshirani, and J. Friedman, The elements of statistical learning, Springer Series in Statistics, Springer, 1st edition, 2001.

[13] CAVIAR test case scenarios. URL http://homepages.inf.ed.ac.uk/rbf/CAVIAR/