21st International Conference on Pattern Recognition (ICPR 2012)
November 11-15, 2012. Tsukuba, Japan

Unsupervised Multi-Target Trajectory Detection, Learning and Analysis in Complicated Environments

Hong Liu and Jiang Li
Key Laboratory of Machine Perception and Intelligence
Peking University, Shenzhen Graduate School, P.R.China
E-mail: {hongliu, lijiang2010}@pku.edu.cn

Abstract

Trajectory analysis is very important to human behavior-analysis for video processing based smart surveillance systems. It has a challenge that human trajectory has no prior model and needs to online learning and updating, while interaction between targets complicates the problem. This paper describes a novel integrated framework for multiple human trajectory detection, learning and analysis in complicated environments. First a modified feature-spatial representation (MFSR) for Cam-Shift tracking algorithm is proposed to obtain trajectories. Then, a piecewise multilevel learning method is adopted to learn the trajectory patterns by using spectral clustering and Hidden Markov Model. Finally a cascade detector is established for anomaly analysis based on learning information, which allows obviously abnormal trajectories to be quickly deviated from normality. Our framework is demonstrated good results by lots of experiments and can be applied in further selective video analysis.

1. Introduction

Nowadays social unstable factors are increasing and people's security awareness is enhancing, surveillance systems are already increasingly commonly used everywhere. Also functions of the surveillance system have changed from some original video signal process to achieving automatic detection, target tracking, and the relevant behavior analysis. Trajectory analysis has become a hot research field in recent years. It has been applied mainly for vehicles and pedestrians research to get the understanding of the behavior or semantic scene[1][2].

The common method of trajectory analysis can be divided into template matching based method [3] or clustering based method, which is the main method. In general procedure, the obtained trajectories are clustered into some classifications trained for the anomaly analysis. Some methods have been used such as improvement AVQ [4], similarity measures[5] and MDS[6].

Unfortunately, the methods proposed above just considered simple behaviors and all of them have a prerequisite that the trajectories are easily obtained. Obviously, interaction between targets, which is very common in real world, should not be ignored under the widely used single-camera surveillance system. There is not an integrated framework for trajectory analysis from trajectory obtaining to anomaly detecting.

In this paper, we present a novel method aiming at the multi-target trajectory processing in complicated environments using a single camera.

Figure 1 shows the framework of our method, which is constituted of three blocks describing the detection, learning and analysis of the multi-human trajectories successively. First, blue block obtains the multiple human trajectories using a modified feature-spatial representation (MFSR) for Cam-Shift tracking algorithm. In red block，an innovative algorithm is formed to model human trajectory called the piecewise hierarchical learning algorithm, which is an offline training procedure. At last, the yellow block shows how the abnormal trajectories are detected online by the maximum likelihood estimation, where the cascade classifier is used for decreasing calculation. In the following sections, more details of the framework are provided.

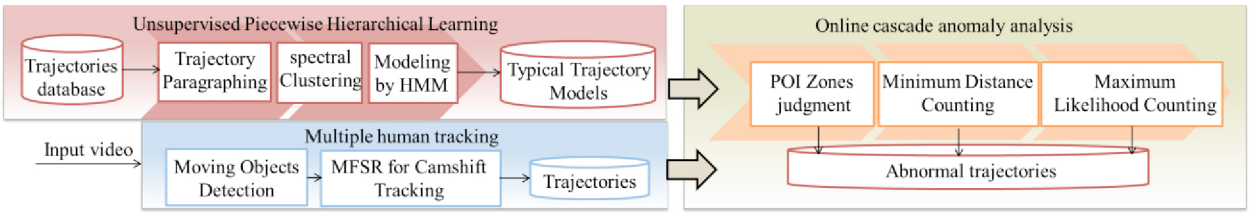


Figure 1. General framework for trajectory detection, learning and analysis. Trajectories are obtained in tracking phase, clustered and modeled during the learning phase. In online analysis phase, the abnormal trajectories are detected.

2. Modified feature-spatial representation (MFSR) for Cam-Shift Tracking

The first procedure is to obtain the trajectories with the help of tracking in multi-target environment. In this issue partial occlusions will be one of the critical challenges. Occlusions disrupt the appearance of tracked targets and complicate the tracking problem. Here a MFSR for Cam-Shift tracking is formed to solve the interaction problem between targets. The adaptive feature-spatial representation (FSR) is demonstrated by Liu et al[7], solving some problems in a single target tracking such as target deformations and partial occlusions.

2.1. Overview of FSR

Obviously, if a part of a human body is occluded by others, the part's reliability must be smaller than the rest part. In a word, the values of pixels at the same location may give different contributions to the target overtime. In FSR, a target is divided into M blocks $\{B_j, j=1,2,3,\ldots,M\}$. And the corresponding weights of each block are different and computed by a specific spatial weighted kernel. By FSR, the block will be ignored by the Mean-Shift tracker when it is occluded, while reliable located ones will get larger weights. The weight function $\psi(x)$ is joined to make it adaptive.

FSR has brought one situation of increasing the calculation and cannot fit for real-time analysis in multi-target tracking, because all targets will be divided into blocks simultaneously. Besides, the color representation is not enough to deal with the situation of some people wearing clothes in similar color.

2.2. Formulation of MFSR

In our MFSR method, multi-scale segmentation is proposed to solve the calculation problem. Meanwhile we adopt the idea of multi-feature fusion[8] which is suitable for multiple human scenes.

A target candidate centered at location y is characterized by its color histogram:

$$\hat{q} = \{\hat{q}_u\}_{u=1,2,\ldots m} \quad where \sum_{u=1}^{m} \hat{q}_u = 1 \qquad (1)$$

$$\hat{p}(y) = \{\hat{p}_u(y)\}_{u=1,2,\ldots m} \quad where \sum_{u=1}^{m} \hat{p}_u(y) = 1 \qquad (2)$$

The color probability distribution map $P_{color}(x,y)$ is got in HSV color space, then the most probable motion $P_{motion}(x,y)$ is detected by parameter estimation and compensation for affine motion between former and later frames. So probability distribution map can be demonstrated as $P(x,y) = P_{color}(x,y) * P_{motion}(x,y)$.

Each target is first segmented into M_1 blocks in coarse scale, called scale *1*. If occlusion situation is serious that all blocks happen, the segmentation on it will be operated in finer scale. For scale k, each block is $\{B_{kj}, j=1,2,3,\ldots,M_k\}$. Here k is determined by the degree of occlusion δ_{degree}. The more serious the occlusion, larger δ_{degree} is , and the more blocks are got. Since the occurring probability of the serious occlusion situations is relatively small, this modification will reduce the computation time, and make the tracking procedure selective attentive.

Now the $\psi(x)$ in scale k on a target can be formulated as the sum of weights of x in all the $\prod_1^k M_k$ blocks:

$$\psi(x) = \prod_{j=1}^{M_k} \phi_j \kappa\left(\left\| \frac{x_i - y^{(t)} - \overline{z}_j}{h_j} \right\|^2\right) \qquad (3)$$

And our candidate MFSR in scale k becomes:

$$\hat{p}_u(y) = C_p \sum_{i=1}^{N} \delta[b(x_i) - u] \prod_{j=1}^{M_k} \phi_j^{(u)} \kappa\left(\left\| \frac{x_i - y^{(t)} - \overline{z}_j}{h_j} \right\|^2\right) \qquad (4)$$

where $\overline{z}_j$ specifies the center of block j, $h(j)$ denotes the range of decay for the j^{th} block. And $\{x_i\}_{i=1\ldots N}$ are normalized pixel locations in the target candidate region. Function $b(.)$ maps a pixel to its feature value. Kernel $\kappa(.)$ is an original formulation[6].

And ϕ_j is determined by the target region and the candidate region, used to smooth the sudden variation of the two region.

2.3. Cam-Shift tracking based on MFSR

The Mean-Shift vector can be derived as follows:

$$y^{(t+1)} = \frac{\sum_{i=1}^{n} \omega_i \prod_{j=1}^{M_k} \phi_j g\left(\left\| \frac{x_i - y^{(t)} - \overline{z}_j}{h_j} \right\|^2\right)\left(\frac{x_i - \overline{z}_j}{h_j^2}\right)}{\sum_{i=1}^{n} \omega_i \prod_{j=1}^{M_k} \phi_j g\left(\left\| \frac{x_i - y^{(t)} - \overline{z}_j}{h_j} \right\|^2\right)\left(\frac{1}{h_j^2}\right)} \quad (5)$$

where $g(.) = -\kappa(.)$.

Then whole Cam-Shift algorithm can be described in Table 1.

Table 1. Cam-Shift algorithm with MFSR

Initialization: For each target: build the probability distribution map of target and set the parameters of Mean-Shift searching window.

Iteration: For $t = 1,\ldots,T$

1. Compute candidate $P(x, y)$.
2. Segment target in M_1 blocks $\{B_{1j}, j = 1,2,3,\ldots,M_1\}$.
3. Determine the k by δ_{degree} and re-segment target to M_k blocks.
4. Run Mean-Shift process and get the new information of searching window, which is the initial value in the next frame.

3. Unsupervised learning

A trajectory $R_i = \{r_t\}$, where $r_t = \left[x_t, y_t, (u_t, v_t)\right]^T$, demonstrates all activity history, which contains some cues such as location, velocity, time. So in our piecewise hierarchical learning algorithm, the trajectories are learned piecewise by two levels called Spatial level and Temporal level.

3.1. Piecewise algorithm

The first procedure is to make the trajectories in segmentations by locating the points of interest (POI) [9] in the image plane. The three types of POI: entry, exit, and stop, actually mean the important positions in a scene. The POI can be obtained by a 2D mixture of Gaussian zone modeling process.

After this operation we can get the origin and destination information for trajectories and segment them into k parts by the stop nodes. So the trajectory R_i is described as $\{R_{ik}, k = 1,\ldots,K\}$.

3.2. Two-level hierarchical learning

Spatial level can evaluate the trajectory prototypes by synthesizing trajectories in models while ignoring the intrinsic variations of each individual one. Spectral clustering is used in this level for classifying the trajectories in several typical prototypes. The similarity matrix $W = \{w_{ij}\}$ is founded from the trajectory distance D_{ave} by using the Gaussian kernel equation:

$$w_{ij} = e^{-D_{ave}^2(R_{ik}, R_{jk})/2\sigma^2} \quad (6)$$

where $D_{ave}\left(R_{ik}, R_{jk}\right)$ is the distance between trajectory R_i and R_j, which can be defended as:

$$D_{ave}\left(R_{ik}, R_{jk}\right) = \left(D\left(R_{ik}, R_{jk}\right) + D\left(R_{jk}, R_{ik}\right)\right)/2 \quad (7)$$

As the lengths of trajectories are different because of the different duration of each trajectory, a kind of normalization technique is integrated to solve this problem.

The next step is to compute the normalized Laplacian matrix:

$$L = 1 - D^{-1/2} W D^{-1/2} \quad (8)$$

with the D the diagonal degree matrix with elements the sum of the same row in W. A new $N \times W$ matrix, U is built using the first K eigenvectors of L as columns. Then cluster the rows of U using k-means.

Temporal level provides a more accurate description about the manner in which the targets are moving. And the probabilistic models for the human trajectories are got by using of Hidden Markov Model (HMM).

The HMM $\lambda_n = (A_n, B_n, \pi)$, where $n = 1,\ldots,N_r$, means the number of the activities one trajectory has, is represented after some learning from the past trajectory training database.

The $Q \times Q$ state transition probability matrix $A = \{a_{ij}\}$

$$\text{where } a_{ij} = p(q_{t+1} = j \,|\, q_t = i) \quad (9)$$

The observation probability distribution $B = b_j(f)$ where

$$b_j(f) = G(f, \mu_j, \sum_j) \quad (10)$$

stands for the Gaussian flow f distribution of each of the $j = 1,...,Q$ states of unknown mean μ_j and cowariance $\sum_j$.

This HMM is a left-right hidden Markov model, in which $a_{ij} = 0$, for $j < i$. And π can be defined as

$$\pi_0(j) = \frac{1}{C} e^{-\zeta_p j} \quad j = 1,...,Q \quad (11)$$

C is a normalization constant for ensuring the valid probabilities. ζ_p is a user defined weight.

The probabilistic models are finalized after both the transition probabilities *A* and the action states which define *B* is learned.

With this process the observation sequences of trajectories can be represented in details.

4. Online anomaly analysis

Piecewise hierarchical learning information is used to establish a cascade abnormal trajectory detector combining increasingly more complex classifiers, which is established by the obvious level of the abnormal trajectory [10]. It allows abnormal trajectories to be quickly detected while spending more computation on promising normal-like trajectories.

Table 2. Online cascade detecting algorithm

Initialization

For each target, set classifier $c_j (j = 1,2,3)$ and define the value of each threshold.

$$c_1 = \begin{cases} 1 & \text{the value of the feature } f_1 \text{ fits the classifier} \\ 0 & \text{otherwise} \end{cases}$$

$$c_2 = \begin{cases} 1 & D_{ave\min} \le \delta_{D_{ave}} \\ 0 & \text{otherwise} \end{cases}$$

$$c_j = \begin{cases} 1 & Lik\left(R_{ik} \mid \lambda_{n\max}\right) \ge \delta_{Lik} \\ 0 & \text{otherwise} \end{cases}$$

Iteration

for $k = 1 : K$ **do**

  Input trajectory R_{ik}

  for $j = 1,2,3$ **do**

    Compute feature f_j ; Compute c_j ;

    If $c_j = 0$ then output R_i as an anomaly

  end

end

When persons are walking into the monitoring area, MSFR tracking procedure will obtain their trajectories and a piecewise process will begin. The first detector judges ever whether its POI nodes are right or not. And the trajectories with wrong POI are outputted the detector.

The rest trajectories will enter the second detector which using minimum distance as its feature. If $D_{ave\min} > \delta_{D_{ave}}$, this trajectory is judged as an abnormal one. The last detecting process is related to maximum likelihood estimation:

$$\Lambda^* = \arg\max_n P(R \mid \lambda_n) \tag{12}$$

If maximum likelihood $Lik\left(R_{ik} \mid \lambda_{n\max}\right)$ is smaller than the threshold δ_{Lik}, the trajectory R_i is considered as an anomaly. And Table 2 presents the cascade detecting procedure.

5. Experiments and Discussions

Our algorithm is tested with 5 real videos in complicated environments: a few people with interaction appear simultaneously in an indoor ATM room. And we use almost 9000 frames for the normal trajectory model testing. In the experiment, the degree of occlusion $\delta_{\deg ree}$ is set by the number of blocks whose weights are zero in last layer, and each threshold is obtained in former experiment.

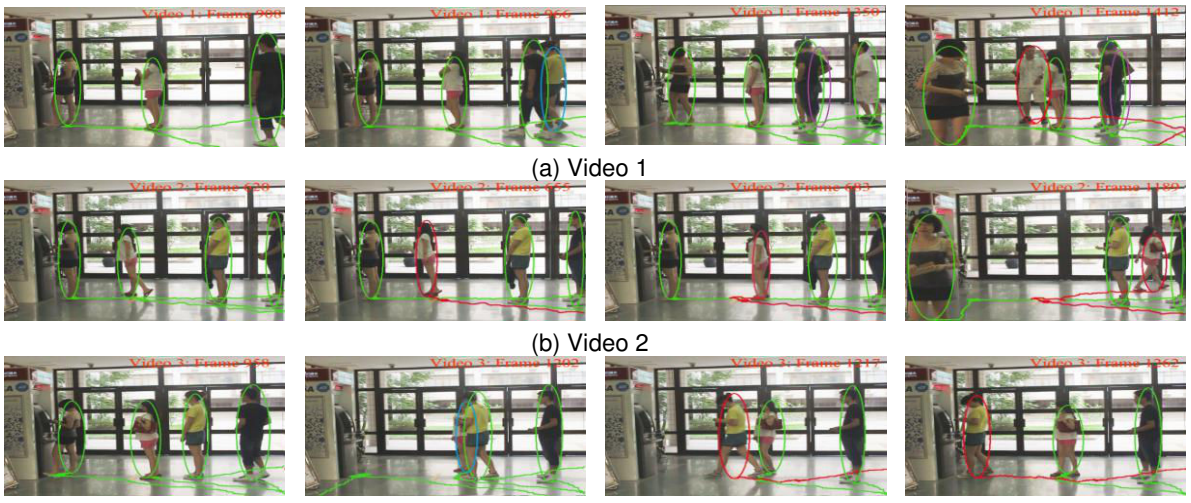
Figure 2 shows different kinds of anomaly detection results in three videos. Each abnormal target was marked by red circle as soon as been detected. Video 1 shows a prowler walking around the line. Video 2 detects the activity of peeping and leaving before using the ATM. And video 3 describes a situation of jumping the queue.

In Figure 2, the frame 966 in video 1 has an occlusion situation, the target in blue circle was blocked by her partner. Frame 1358 and frame 1412 in video 1 show the more serious occlusion situations. In these two scenes, we can hardly see the girl behind the man wearing black T-shirt, but trajectories are still obtained continuously by using the MFSR Cam-Shift tracking process. But the FSR method cannot keep rapid tracking thus the trajectory analysis cannot proceed.

Table 3. Comparison with original FSR

Algorithm	Time Efficiency/ms					Precision
	V1	V2	V3	V4	V5	
MFSR	131	126	129	141	135	94.5%
FSR	189	132	144	148	136	83.9%

It can be found that our algorithm shows a better performance than the original FSR tracker. Table 3 presents the average time required for each target and precision of two methods, proving our algorithm can guarantee an online analysis.



(a) Video 1

(b) Video 2

(c) Video 3

Figure 2. The experimental results. Red trajectories are abnormal and green ones are normal. Blue circle is the deformational target and purple one represents the more serious situation.

6. Conclusions

In this paper we describe a novel integrated framework from human trajectory detection to anomaly analysis in complicated environments. It is worth mentioned that we integrate the MFSR tracking algorithm solving interaction problem between targets and establish a cascade anomaly detector using the information by a piecewise two-level unsupervised learning procedure. Our lots of experiments make good presentations for real-time trajectories obtaining of multi-target and show that our algorithm achieves rapid anomaly detecting in complicated environment. Since the framework is self-learning and unsupervised, the general method can be imbedded in surveillance systems for further human action analysis.

Acknowledgement

This work is supported by National Natural Science Foundation of China(NSFC, No.60875050, 60675025), National High Technology Research and Development Program of China(863 Program, No.2006AA04Z247), Scientific and Technical Innovation Commission of Shenzhen Municipality (No.JC201005280682A, CXC201104210010A).

References

[1] X. Song, X. Shao, H. Zhao, J. Cui, R. Shibasaki and H. Zha, *"An online approach: Learning-Semantic-Scene-by-Tracking and Tracking-by-Learning-Semantic-Scene"*, CVPR June.2010, pp.739-746.

[2] H. Yang, L. Shao, F. Zheng, L. Wang and Z. Song, *"Recent advances and trends in visual tracking: A review"*, Neurocomputing 74 (2011), pp. 3823–3831.

[3] S. Calderara S, R Cucchiara, and A. Prati, *"A Distributed Outdoor Video Surveillance System for Detection of Abnormal People Trajectories"*, ISDSC Sept. 2007, pp. 364 – 371.

[4] A. Mecocci and M. Pannozzo, *"a completely autonomous system that learns anomalous movements in advanced video surveillance applications"*, ICIP Sept.2005, pp. II - 586-9.

[5] D. Makris and T. Ellis, *"Learning semantic scene models from observing activity in visual surveillance"*, Transactions on SMC June. 2005, pp. 397 - 408.

[6] N. Suzuki, K. Hirasawa, K. Tanaka, Y. Kobayashi, Y. Sato and Y. Fujino, *"Learning motion patterns and anomaly detection by Human trajectory analysis"*, ICSMC Oct. 2007, pp. 498 - 503.

[7] Y. Shi, H. Liu, Y. Liu and H. Zha, *"Adaptive feature-spatial representation for Mean-Shift tracker"*, ICIP Oct.2008, pp. 2012 – 2015.

[8] Z. Li, H. Liu and C. Xu, *"Real-time human tracking based on switching linear dynamic system combined with adaptive Mean-Shift tracker"*, ICIP Sept. 2011, pp. 2329 – 2332.

[9] B.T. Morris and M.M Trivedi, *"Trajectory Learning for Activity Understanding: Unsupervised, Multilevel, and Long-Term Adaptive Approach"*, PAMI Nov. 2011, pp. 2287 – 2301.

[10] P.Viola and M. Jones, *"Rapid object detection using a boosted cascade of simple features"*, CVPR2001, pp. I-511 - I-518 vol.1.