ISSN 0868 - 3980

Tạp chí KHOA HỌC & CÔNG NGHỆ CÁC TRƯỜNG ĐẠI HỌC KỸ THUẬT

JOURNAL OF SCIENCE & TECHNOLOGY TECHNICAL UNIVERSITIES

B

No.83
2011

CONTENTS

MULTIPLE OBJECTS TRACKING FOR VISUAL SURVEILLANCE

PHÁT HIỆN VÀ THEO DÕI NHIỀU ĐỐI TƯỢNG TRONG VIDEO
ỨNG DỤNG TRONG THEO DÕI GIÁM SÁT

Tran Thi Thanh Hai
Hanoi University of Science and Technology

Do Nguyen Trung
Petrolimex Engineering Company - Hanoi

ABSTRACT

Object tracking becomes a problem that interests more and more researchers due to its wide applications in reality. In monitoring application where we need to detect and track multiple people in a scene, multiple object tracking is necessary. The problem of multiple object tracking is much more complex than single object tracking because we must do an additional step: data association as well as manage the apparition and disappearance of the objects. In this paper, we proposed a framework for multiple objects tracking using Kalman filter with the data association using histogram based similarity measurement. This method is simple but efficient; it permits tracking multiple people in real-time so can be applied in real applications.

Keywords - Object Tracking, Data Association, Kalman Filter, Histogram

TÓM TẮT

Theo dõi đối tượng trong video là một hướng nghiên cứu có ý nghĩa quan trọng trong lĩnh vực thị giác máy tính bởi các ứng dụng đa dạng của nó trong thực tế. Trong bài viết này, chúng tôi đề cập đến bài toán theo dõi nhiều đối tượng ảnh phục vụ cho các ứng dụng theo dõi giám sát. Bài toán này phức tạp hơn bài toán theo dõi một đối tượng ảnh do có nhiều quan sát nên phải thực hiện gán các quan sát đối tượng cũng như quản lý sự xuất hiện hay biến mất của các đối tượng trên khung hình. Bài báo đề xuất sử dụng bộ lọc Kalman cho việc dự báo chuyển động của đối tượng và thước đo so sánh sự sai khác của các quan sát dựa lược đồ xám của ảnh. Phương pháp này đơn giản nhưng hiệu quả, cho phép theo dõi nhiều đối tượng ảnh thời gian thực, vì thế hoàn toàn có thể ứng dụng trong thực tế.

I. INTRODUCTION

Object tracking in video is an essential requirement for systems employing one or more camera to interpret the environment. Nowadays we can see a lot of applications enabled by object tracking such as security and surveillance, video abstraction, traffic management, video editing, interactive game, sport performance evaluation, etc [1, 7, 10].

Depending on applications, single object tracking and multiple object tracking will be considered. In general, multiple object tracking appears more frequently in reality. For example, at metro/airport station, one would like to monitor more than one person moving in the scene. On the road, the traffic management requires to detect and track not only one but more vehicles at the same time.

Multiple object tracking is much more complex than single object tracking because there are a lot of objects then we need to assign observation to object (data association problem). In addition, object tracking in video must face several difficulties: abrupt object motion, changing appearance patterns of object and scenes, non- rigid object structure, occlusion and camera motion, etc. Beside, object tracking requires to work in real-time while the number of objects to be tracked can be a lot (tracking in a crowded scene).

Assuming we have a video containing moving objects to be tracked. Tracking aims to follow an object during time, more specifically to find state (e.g. position, velocity) of the same object in the next frame. The problem of multiple object tracking consists of following steps:

i. *Prediction*: Predict the position of each object in the next frame
ii. *Detection*: Determine the observations surrounding predicted regions

iii. *Data association*: Assign observation to object

iv. *Update* state of the object

To initialize the tracking as well as to determine observations, object detection is necessary. In general, many works assume that object moves in a static scene captured by a static camera. Therefore, the object detection for the tracking is usually performed by background subtraction technique [12]. However, this hypothesis is not always true so one needs a more robust object detector. Recently, some robust object detection methods have been proposed and obtained very good results in precision term [8]. In these methods, objects are represented by feature descriptors computed on several saliency positions. The descriptor could be encoded as a visual word in visual dictionary and the problem of multi-dimensional feature classification could be resolved by techniques like Neural Network, SVM or Adaboost classifier [6]. The main problem of these approaches is they are very time consuming. To be able to apply in real-time applications, one needs to speed up the object detection.

The *prediction* aims to predict state of the object in the next frame using maximum measured information from the past to current time. One important question is how to model movement of the object. The movement in reality could be linear, with constant velocity but could be very complex. Kalman filter, an extended of Least Squares Estimation method, assumes that state variables of the object satisfy Gaussian distribution. It has been widely applied for object tracking and obtained good results [2]. When the hypothesis is not true, the particle filter is used [9]. However, particle filter based solution is much more time-consuming than Kalman filter.

Note that the Kalman filter and particle filter assume a single measurement at each time instant, that is, the state of a single object is estimated. Tracking multiple objects requires a joint solution of data association and state estimation problems.

The simplest *observation - object assignment* is to use the nearest neighbor approach. However, if the objects are close to each other, there is always a chance that the observation-object correspondence is incorrect. There exist several statistical data association techniques to resolve this issue such Joint Probability Data Association Filter (JPDAF) or Multiple Hypothesis Tracking (MHT) [4]. A main problem of JPDAF is it is performed on a fixed number of objects; therefore serious errors can arises if there is a change in the number of objects in the scene. MHT can tackles to this problem and also occlusion problem. However, MHT is computationally exponential both in memory and time because it makes the association in a deterministic scene and exhaustively enumerates all possible associations.

In this paper, we propose a framework of multiple object tracking consisting of 3 components: i) *motion detection* using adaptive background subtraction, so it is more robust than traditional background subtraction; ii) *Kalman filter* as position prediction; iii) *histogram* based data association. As each component is carried out by a simple but efficient technique, then the problem of computational time can be handled. We have experimented our system with real scene that shows a good multiple people tracking results.

In the following, we will present in detail each component in the framework (section II). We expose the experimental results in the section III and give some conclusions in section IV.

II. PROPOSED TRACKING FRAMEWORK

As previous speaking, the multiple objects tracking framework consists of 3 principal components: object detection; state prediction and correction; data association as illustrated in the figure 1. These components are within 2 main blocks: Object detection and Object Tracking that we detail in the following sub-sections.

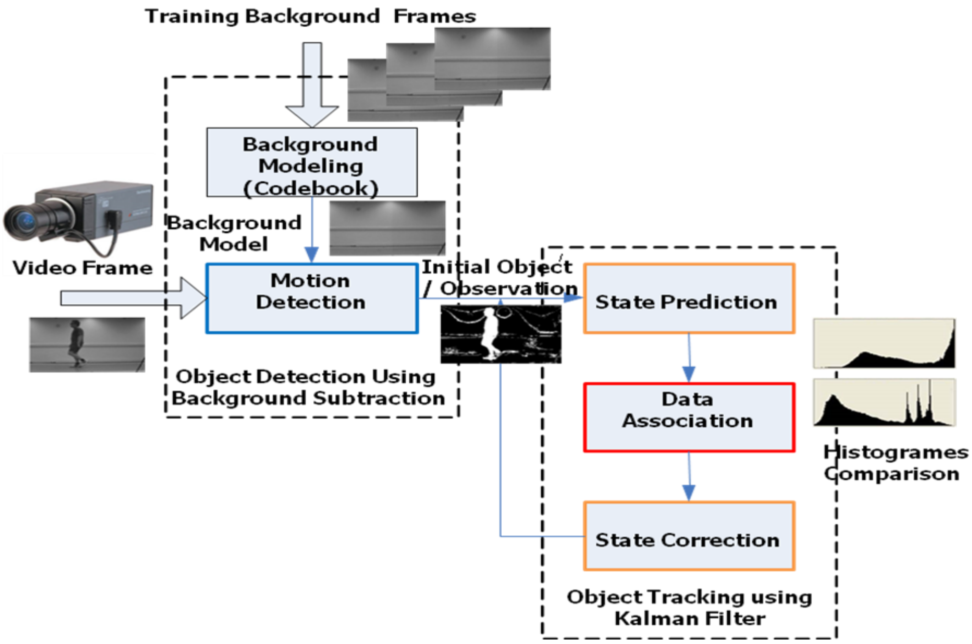

Figure 1. Proposed framework of multiple object tracking.

2.1 Object detection

1. Background Modeling

The object detection assures an automatic initialization of the tracker as well as provides observations for data association. In our context, the camera is fix but scene can contain moving background (like waving trees) and illumination variations. To handle to this, we proposed to use segmentation technique based on codebook [Kim05].

The main idea of the method is to construct background model from long training sequences. Each pixel of background will be encoded by a codebook composed from several codeword computed from color distortion metric together with brightness bounds. By this way, it allows to build adaptive and compact background model that captures the structural background motion in a long time and the ability to cope with local and global illumination change. The reason that we choose this algorithm for background modeling is that this method is experimentally shown to be more efficient in time/memory and precision, meaning some moving elements in the background is considered as background. For technical detail, see the original paper [5].

2 Moving object detection

Once the background model is built, given each video frame, the moving objects detection is carried out by differencing the current image with the background model. To remove noises, we threshold the different image. Morphological operators are then used, followed by connected component analysis to group pixels into blobs. The small blobs corresponding to the noise are skipped.

2.2 Object tracking

Object tracking consists of predicting position and observation-object assignment.

Kalman filter for prediction and correction

Kalman filter has been studied since 1960 and extensively used in the vision community for tracking by its efficiency in real applications. By this reason, we proposed to use Kalman filter. Kalman filtering is composed of 2 steps: prediction and correction. The prediction step uses the state models to predict the new state of the variables. The correction step uses the current observation to update the object's state. Two steps of Kalman filter is presented as in the figure 2.

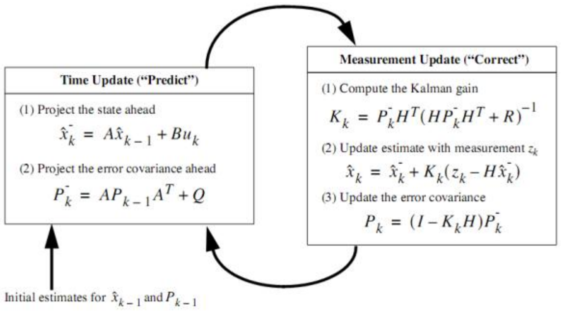

Figure 2. Two steps of Kalman filter

In this figure, $\hat{x}_k^-$ is a priori state estimate at step $\boldsymbol{k}$ given knowledge of the process prior to step $\boldsymbol{k}$. $\hat{x}_k$ is a posteriori state estimate at step $\boldsymbol{k}$ given measurement $\boldsymbol{z_k}$. P_k^- and $\boldsymbol{P_k}$ are a priori/posteriori estimate error covariance. $\boldsymbol{K}$ is a gain filter that minimizes the a posteriori error covariance. $\boldsymbol{A}$ is the relation between the state at the previous time step $\boldsymbol{k-1}$ and the state at the current step $\boldsymbol{k}$; $\boldsymbol{B}$ relates the optional control input $\boldsymbol{u_k}$ to the state $\boldsymbol{x_k}$.

2.3 Observation-object assignment

For each time $\boldsymbol{k}$, there are a lot of objects to be tracked and observations (moving object detected). The Kalman filter will use the observation corresponding to the object to predict and update the state of the object in the next frame. The question here is which observation corresponds to which object?

We propose to compare each object with all observations and choose the most similar one as candidate for assignment. To compare observation - object, we describe each one by an appearance based descriptor. There are a lot of descriptors proposed in the literature, but histogram based one is simple and as between two frames, the movement of the object is not significant, the comparison of histograms is sufficient to give reliable matching result. In addition, we normalize the histogram so that the similarity measurement is independent to scale change.

In our implementation, the number of bins is chosen experimentally. We found that the bin number equaling to 128 gives the best rate of correspondence.

The difference between histograms is computed base on Chi-square measurement. Although the computation of *Chi-square* is a little more time-consuming than *Correlation* and Intersection but it gives more precise results of matching [3]. The *Chi-square* distance is defined as in equation (1):

$$d_{\text{chi-square}}(H_1, H_2) = \sum_i \frac{(H_1(i) - H_2(i))^2}{H_1(i) + H_2(i)} \qquad (1)$$

To make observation - object assignment, for each object, we find the observation having the most similar histogram w.r.t the histogram of this object. We note that for each object, we don't consider all detected observations but only ones within bounded region centered by the object center. This has twofold: i) it reduces the computational time to compute all pairs of object-observations like in MHT; ii) it removes outlier matching because an object cannot move too far from given position.

I. EXPERIMENTS

3.1 Dataset for experiment

To evaluate our framework, we do experiments with videos containing people walking together. We use two videos (resolution: 384x288) coming from CAVIAR project [14] and two other videos (resolution: 320x240) that we capture with webcam (not good quality of image). In these videos, there are one or two people walking in the scene at the same time. People can walk in the same or cross-direction. The ground truth is determined by hand. Given the ground truth, the performance of tracking system can be evaluated by computing *Recall* and *Precision* measures that are defined as following:

$$\textit{Precision} = \frac{\textit{Number of correct corres.}}{\textit{Number of established corres.}} \qquad (2)$$

$$\textit{Recall} = \frac{\textit{Number of correct corres.}}{\textit{Number of actual corres.}} \qquad (3)$$

in which *actual corres.* is the number of correspondences in the ground truth.

3.3 Experimental Results

We perform experiments in PC - 2.83 GHz. The tracking system works at 20 fps with experimental videos. The recall is 88 % and the precision is about 94%. The following figures show several examples of good and bad tracking. In figure 3, at left shows the different image obtained from background subtraction module. At center, moving objects are detected using noise removing and connected component analysis. At right is the result of matching. Each tracked object is colored by a distinct color. We found that the codebook algorithm models well background and the connected component analysis gives "clean blobs" to be considered as initial objects to be tracked or observations.

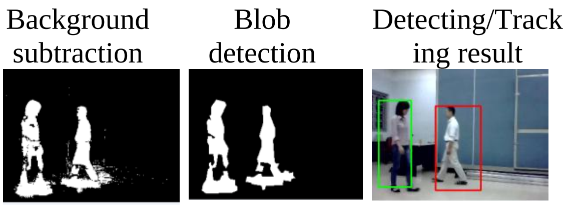

Figure 3. Different steps of multiple object tracking.

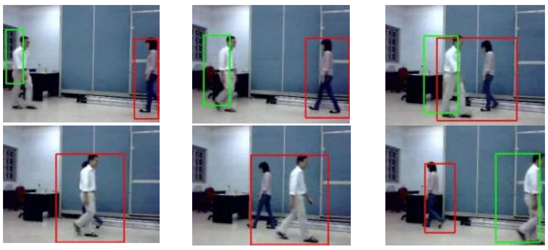

Figure 4. Results of tracking at different frames (from top to bottom, left to right: 59, 62, 67, 70, 72, and 77).

Figure 4, 5 shows series of frames in video that we do the tracking. Each object is identified by a specific color. The consistence of color over time demonstrates the good tracking of the object. We found that when two objects are close together, only one observation is detected, in this case, our algorithm considers that two objects share the same observation. The observations will appear separately when two objects go far from each other.

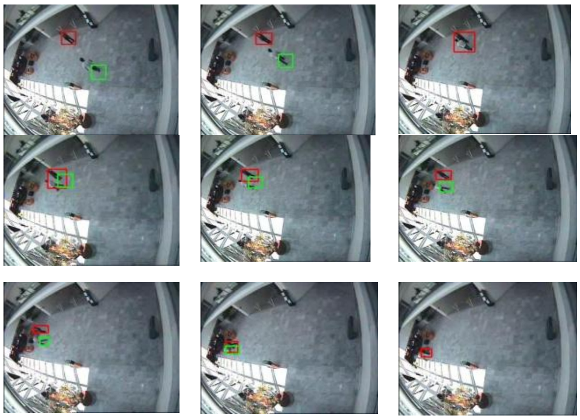

Figure 5. Result of object tracking with videos coming from CAVIAR project at different frames (from top to bottom, left to right: 5, 9, 11, 19, 21, 25, 28, 31, and 37).

II. CONCLUSIONS

In this paper, we presented a fully automated framework for multiple object tracking. Our method based on simple techniques as Codebook based background subtraction, Kalman filter and data association using histogram comparison, but has been shown quite efficient.

We validated our framework with real data, and obtained good results. The experiments show that: i) codebook based background subtraction is robust in case of change in illumination. ii) in people tracking application, Kalman filter is sufficient to determine their trajectories correctly. In the future, we will test our framework with more complex movements of objects in unconstrained environment. We will also evaluate the precision of object trajectory estimation.

Acknowledgement

The research leading to this paper was supported by the National Project ĐTĐL.2009G/42 "Study, design and develop smart robots to exploit multimedia information", under grant 42/2009G/HĐ-ĐTĐL. We would like to thank the project and people involved in this project.

REFERENCES

1. J. Berclaz, F. Fleuret and P. Fua, "Robust People Tracking with Global Trajectory Optimization", In Proc. of In Conference on Computer Vision and Pattern Recognition, 2006.
2. S. Blackman and R. Popoli, "Design and Analysis of Modern Tracking System", Artech House Publishers, 1999.
3. G. Bradski, A. Kaehler, "Learning OpenCV", O'Reilly Media Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2008.
4. I, Cox, and S. Hingorani, "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking". IEEE Trans. Patt. Analy. Mach. Intell. 18, 2, 138–150, 1996.
5. K. Kim T. H. Chalidabhongse, D. Harword and L. Davis, "Real-time foreground-background segmentation using codebook model, Real-time Imaging", Elsevier, 2005.
6. S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques", Informatica, 31 (2007), 249-268.
7. K. Okuma, A. Taleghani, N. de Freitas and I. Iittle and D. Lowe, "A boosted particle filter: multitarget detection and tracking", ECCV2004.
8. J. Ponce, M. Hebert, C. Schmid, A. Zisserman, "Toward Category-Level Object Recognition", Lecture Notes in Computer Science, Vol. 4170, 2007.
9. B. Ristic, S, Arulampalam, N, Gordon, "Beyond the Kalman Filter: Particle filters for Tracking Applications", Artech House Radar Library, 2004.
10. A. Yilmaz, O. Javed, M. Shah, "Object Tracking: A survey", ACM Inc., 2 Penn Plaza, Suite 701, NewYork, NY10121-0701, 2006.
11. T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environment", In IEEE Proc. of Computer Vision and Pattern Recognition, 2004.
12. C. Wren and A. Azarbayejani, A. Pentland, "Pfinder: Real-time tracking of the human body". IEEE Trans. Patt. Analy. Mach. Intell. 19, 7, 780–785, 1997.
13. http://homepages.inf.ed.ac.uk/rbf/CAVIAR/

Author's address: Tran Thi Thanh Hai – Tel: (+84)909621178 - Email: thanh-hai.tran@mica.edu.vn
Hanoi University of Science and Technology
No.1 Dai Co Viet Str., Ha Noi, Viet Nam