社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2006－CVIM－152（18）
2006／1／20

カメラワークを伴ったビデオ映像における動物体追跡

Yang WANG*　　平川 正人+

*島根大学大学院 総合理工学研究科 数理・情報システム学専攻
+島根大学 総合理工学部 数理・情報システム学科
Email: {s049319, hirakawa} @ cis.shimane-u.ac.jp

あらまし　コンピュータ技術や画像処理技術の発達によって、撮影されたビデオから動物体を追跡する研究が注目されている。この技術は監視システムやビデオ検索/編集など様々な分野で活用されている。そこで本論文では単眼移動カメラで撮影された（パン、チルト、ズームを含む）ビデオに対し、コンピュータが自動的に複数動物体を抽出・追跡する手法を提案する。ユーザがマウスドラッグで注目したい動物体を選ぶことによって、コンピュータが自動的にテンプレートを作成し、またカメラのパン、チルト、ズームの操作パラメータを計算する。その後、テンプレートとカメラワークのパラメータを利用して動物体を追跡・抽出する。追跡精度を保持するべく、時間の経過と共にテンプレートを逐次更新する。実験結果により提案手法の有効性を確かめた。

Tracking Moving Objects in a Video Stream with Camera Motion

Yang WANG*　Masahito HIRAKAWA+

*Interdisciplinary Graduate School of Science and Engineering, Shimane University
+Interdisciplinary Faculty of Science and Engineering, Shimane University
Email: {s049319, hirakawa} @ cis.shimane-u.ac.jp

Abstract　Moving object tracking in, for example, surveillance tasks, video retrieval and video editing, is becoming more and more important these days. We present an approach to track more than one object in a panned, tilted and/or zoomed video. In our approach, users select moving objects as tracking targets by dragging a mouse. The algorithm first makes templates of those selected objects, based on their positions and gray-scale distributions. Then the camera motion parameters of pan, tilt and zoom are actively computed. At last the positions of moving objects are determined by referring to the camera motion parameters and the templates of moving objects. During the computation, the templates are modified actively to keep the accuracy of tracking. Preliminary experimental results show that the algorithm performs well.

1.Introduction

Having witness of the success in web camera applications and the appearance of high definition digital video cameras, we believe that digital video media will soon become a part of our everyday life. Unlike still images, video sequences provide more information about how objects and scenarios change over time. Computer vision approaches for object tracking are getting more and more interesting due to their capabilities in getting the information of moving objects.

Much work has been devoted to efficient object identification and tracking. Several algorithms for tracking moving objects across multiple stationary cameras have been proposed recently [1]-[4]. Multiple cameras are commonly used for monitoring large regions of interest. They guarantee a wide coverage of the area and good image resolution, allowing the inference of additional characteristics necessary for activity description and recognition. While tracking objects across multiple cameras is a challenging task, it requires space and time for registration of trajectories recovered from each camera.

Meanwhile, the most widely adopted approach for moving objects tracking with fixed camera is based on background subtraction [5]-[7]. In literature different object and human motion tracking methods have been presented [8], [9]. Some of these methods present medium and high processing cost and do not work well in cluttered scenes with camera motion, while most of the video streams are made from a video camera with motion in our usual situation. The presence of a moving camera makes the problem even more challenging as we have to keep track of the camera motion in order to properly integrate objects' trajectories.

Therefore we present in this paper an efficient method for tracking multiple objects in a complex and cluttered scene with camera motion. The rest of the paper is organized as follows: Section2 gives an overview of the proposed approach. Sections 3 and 4 present how to create a template of moving object for tracking and achieve camera motion estimation, respectively. We will introduce a tracking algorithm in section 5. An experimental result is shown in section 6, and the conclusions are given in section 7.

2. Overview of the Proposed Approach

We are interested in tracking multiple objects in a video which may contain pan, tilt and/or zoom camera motion, since handheld digital camcorders and camera phones are available nowadays, and it is quite natural to place camera motions by means of such equipments.

In our approach, users first select moving objects as tracking targets by dragging a mouse. This means that users can decide which objects should be tracked, allowing them to pay attention only to some of the objects. A template is then created for each of the selected moving objects. Its benefit is that it is not necessary to try identifying which the moving objects are by using a complex algorithm as in [10], [11]. Templates are specified and managed based on their positions and gray-scale distributions.

Meanwhile, camera motion parameters

for pan, tilt and zoom are estimated according to the gray-scale distribution between a frame and its subsequent frame. Here, as you can imagine, the existence of moving objects in the frames degrades the matching performance in camera motion estimation. To avoid this bad influence, the template regions are not taken into account for estimation.

By referring to the result of camera motion analysis, the possible regions of target moving objects in the next frame are estimated. We then determine the specific positions of moving objects through the comparison of color features in the searching regions and their corresponding templates. The newly determined region (i.e., template) displaces the original one actively. Moving object identification and tracking are carried out repeatedly by applying the same procedure.

In this approach, we propose considering the camera motion and visual features of moving objects. The system keeps tracking of objects in a video stream which is taken with a combination of pan, tilt and/or zoom operations. Here it is noted that there is usually a little change between two consecutive frames. The proposed algorithm works even in the situation where an object changes its shape.

3. Templates Formation

An interactive interface that allows the user to manipulate a video frame by frame has been implemented. The user first selects moving objects as tracking targets by dragging a mouse. A rectangle is then drawn along the selected moving object, as shown in Fig. 1.

Fig. 1 Selecting moving objects as tracking targets

For the z-th rectangle region, i.e., the z-th selected moving object, the system extracts a gray-scale value at the position (i, j), which is denoted as $Graytemp\ (z, i, j)$. Assume that the height and width of the rectangle are H_z and $W_{z,}$ respectively. The z-th template is described by:

$$\text{Template(z)} = (H_z\ ,\ W_{z,} \sum_{i=x}^{x2} \sum_{j=y1}^{y2} Graytemp\ (z, i, j))$$

where (x1,y1), (x2,y2) are the start point and the end point of the z-th rectangle. Template(z) is thus a composition of gray-scale values of all the pixels in the z-th rectangle, and its height and width. For every rectangle, the template is generated as explained above.

4. Camera Motion Estimation

In video sequences, the motion in consecutive frames is created by a combination of the motion of a video camera and the movement of objects. Since in this study our interest is to track moving objects, we need to remove the influence caused by camera motion as much as possible.

Camera motion has been studied for a

long time, and there are many papers in the literature such as [12]. In our approach, we compare gray-scale distributions between consecutive frames to estimate the motion of a camera, including pan, tilt and zoom. In order to reduce the noise of computing, we exclude the regions where the selected moving objects are located because the visual features in those regions would change greatly.

Firstly the gray-scale distributions of vertical and horizontal directions of the n-th frame f_n are calculated as

$$projX_n(j) = \frac{1}{A}\sum_{i=1}^{w} Gray(z,\ i,\ j)(i \notin \forall Template(z))$$

$$A = \sum_{i=1}^{w} i(i \notin \forall Tenplate(z))$$

$$projY_n(i) = \frac{1}{B}\sum_{j=1}^{h} Gray(z,\ i,\ j)(j \notin \forall Template(z))$$

$$B = \sum_{j=1}^{h} j(j \notin \forall Template(z))$$

In the above equations, w and h represent the width and height of the frame, respectively, $Gray(z,i,j)$ is a gray-scale value of the pixel at (i, j). $projX_n(j)$ and $projY_n(i)$ give the gray-scale distribution of every vertical and horizontal line, respectively. A and B get the total number of all pixels in a vertical and horizontal line except the moving object region. This scheme is illustrated in Fig.2.

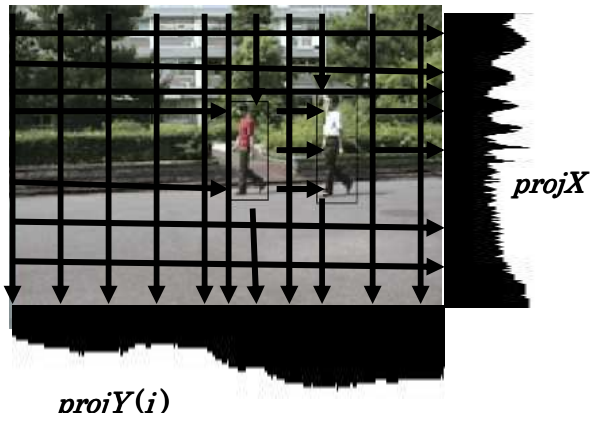

Fig.2 Gray-scale distribution of frame

Next, let us consider a zoom operation and δ be a zooming factor. In this case, $projX_n(j)$ and $projY_n(i)$ defined above are modified to

$$projZX_n(j) = projX_n(j \cdot \frac{h-2\cdot\delta}{h} + \delta)$$

$$projZY_n(i) = projY_n(i \cdot \frac{w-2\cdot\delta}{w} + \delta)$$

since, an original pixel at (i, j) would be found at $(i \cdot \frac{w-2\cdot\delta}{w} + \delta,\ j \cdot \frac{h-2\cdot\delta}{h} + \delta)$ in its subsequent frame.

Then we compare the gray-scale distributions of vertical and horizontal directions between frame f_n and frame f_{n+1}. The difference D_X of vertical gray-scale distribution and the difference D_Y of horizontal gray-scale distribution are computed respectively as

$$D_X = \sum_{j=1}^{h} \{projZX_n(j+\Delta y) - projX_{(n+1)}(j)\}^2$$

$$D_Y = \sum_{i=1}^{w} \{projZY_n(n, i+\Delta x) - projY_{(n+1)}(i)\}^2$$

Here, Δx and Δy are values of pan and tilt of camera motion, respectively. In the above equations, we change the value of δ from $-Z_{max}$ to $+Z_{max}$, Δx and Δy from $-d_{max}$ to $+d_{max,}$ where Z_{max} and d_{max} are the estimated biggest factor of zoom and changing values of pan and tilt between two consecutive frames, respectively. Of course those values can be customized. When Dx and Dy take their minimum values, the corresponding values of δ, Δx and Δy present the factor of zoom, the changing values of pan and tilt of camera motion, respectively.

5. Object Tracking Method

Based on the estimation of camera motion and moving objects' templates, objects tracking process is carried on. According to the value of camera motion, the position of selected moving objects is predicted. This means that, if the selected moving object does not move during the corresponding period, it would appear in the subsequence frame with the coordinates below:

$$zpX1_{n+1} = \frac{(1-\delta)\cdot(zx2_n - zx1_n)}{2} + \Delta x$$

$$zpX2_{n+1} = \frac{(1+\delta)\cdot(zx2_n - zx1_n)}{2} + \Delta x$$

$$zpY1_{n+1} = \frac{(1-\delta)\cdot(zy2_n - zy1_n)}{2} + \Delta y$$

$$zpX2_{n+1} = \frac{(1+\delta)\cdot(zy2_n - zy1_n)}{2} + \Delta y$$

where $zpX1_{n+1}$, $zpY1_{n+1}$, $zpX2_{n+1}$ and $zpY2_{n+1}$, are the predicted coordinates of template z in the frame of $f_{n+1,}$ and $zx1_n$, $zy1_n$, $zx2_n$ and $zy2_n$ are the coordinates of the template z in the frame f_n.

We use zX_{n+1} and zY_{n+1} to express the tracked coordinates of moving object z in the frame f_{n+1}. The differences of the vertical and horizontal gray-scale values, Dzx and Dzy, between the template z and the predicted region in the next frame, are computed as

$$Dzx = \sum_{j=zY_{n+1},jj=zy1_n}^{zY_{n+1}+Hz,zy2_n} (Gray(z,i,j) - Graytemp(z,ii,jj))^2$$

$$Dzy = \sum_{i=zX_{n+1},ii=zx1_n}^{zX_{n+1}+Wz,zx2_n} (Gray(z,i,j) - Graytemp(z,ii,jj))^2$$

Because there is a little change of moving objects between the two consecutive frames, zX_{n+1} and zY_{n+1} are changed one by one pixel from $zpX1_{n+1}-\triangle D_{max}$ to $zpX2_{n+1}+\triangle D_{max}$ and from $zpY1_{n+1}-\triangle D_{max}$ to $zpY2_{n+1}+\triangle D_{max.}$. Here $\triangle D_{max}$ is the maximum moving value estimated to the moving object z, and it can be adjusted. When Dzx and Dzy take the minimum values, the corresponding zX_{n+1} and zY_{n+1} are the coordinates of the tracked moving object z. After that the template z is displaced according to the gray values of the new tracked region. Using this method, all of the selected moving objects are tracked.

6. Results of Experiments

In order to evaluate the performance of the proposed algorithm, we pursued an experiment. We used 34 video streams for the experiment. 9 of them are taken with

any one of the camera motions of pan, tilt, or zoom. 3 video streams include pan and tilt camera motions, 5 with pan and zoom, and 5 with tilt and zoom. And 12 video streams contain all camera motions - pan, tilt and zoom. Also there is more than one moving object in those video streams. Furthermore, some of the videos are taken outdoors and some others are indoors. Bright and dim scenes are included in the data set.

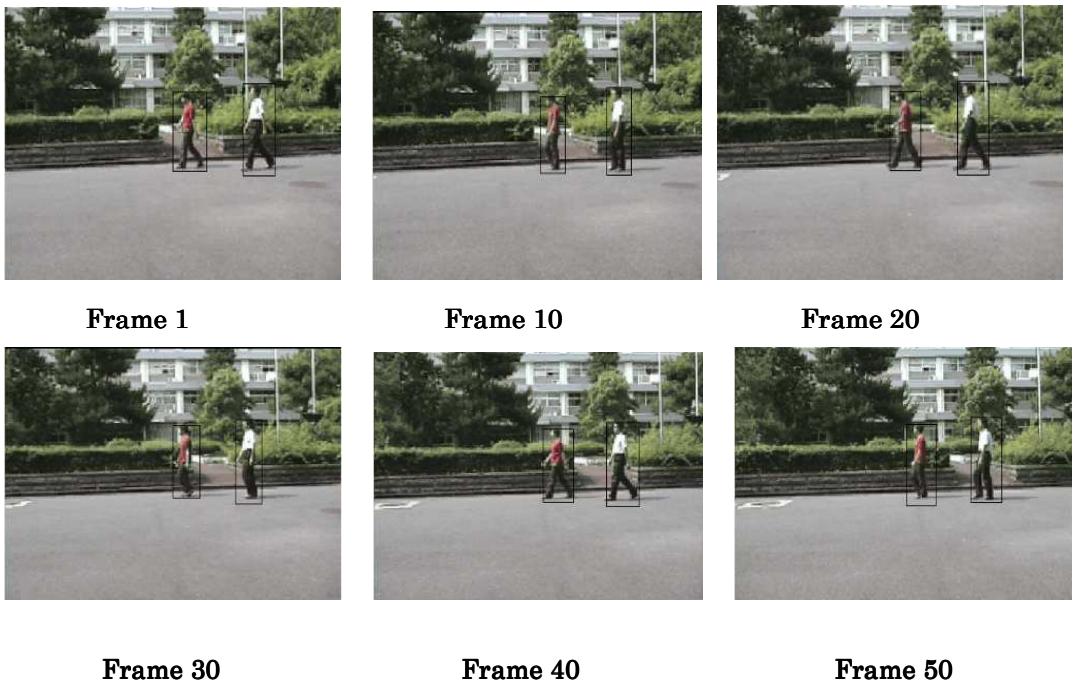

Fig. 3　A result of object tracking (pan)

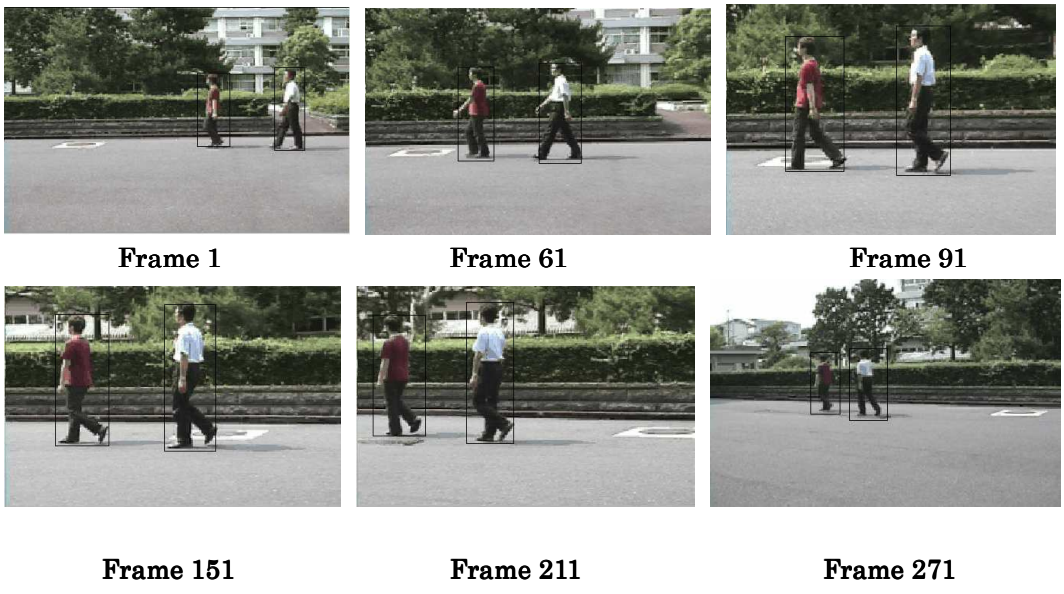

Fig. 4　A result of object tracking (pan+tilt+zoom)

The result of the experiment is summarized in Table 1. Rate is obtained by the number of videos that the selected moving objects are tracked correctly, divided by the number of experimental videos. Fig.3 shows a sample result for the video stream where only pan operation is placed. There are two people and both people are selected as tracking targets. The regions extracted by the system are indicated by rectangular boxes, as in the figure. Fig.4 shows another result, where there are two people as tracking targets and camera motions included are pan, tilt and zoom. As you can see, the system is successful in extracting and tracking objects.

Table 1 Tracking results

Camera Motion	Video Number	Correct Tracking Rate
Pan	3	100%
Tilt	3	100%
Zoom	3	100%
Pan and Tilt	3	100%
Pan and Zoom	5	80%
Tilt and Zoom	5	80%
Pan, Tilt and Zoom	12	83.3%

As indicated in Table 1, though the rate comes down to around 80% for the videos which include zoom and any other camera motion, we succeeded to extract objects properly in the case of one camera motion and pan+tilt motion.

However, it is noted that the algorithm cannot deal with the videos in which objects are overlapped or disappear. Furthermore, if the shape of a moving object changes considerably, the algorithm may fail.

7. Conclusion

We presented in the paper a method of tracking multiple objects in a video stream having camera motions such as pan, tilt, and/or zoom. The method is composed of three phases: creation of moving object templates using gray-scale value as their features, estimation of camera motion of pan, tilt and zoom, and identification of moving objects based on those two results.

Preliminary experimental results demonstrated the effectiveness of the algorithm even in some complicated situations - for example, tracking objects change their shapes, and camera motion includes not only pan and tilt but also zoom.

However, the projective model for camera motion we introduced is somewhat computationally expensive. Also additional studies are needed so that the algorithm can deal with the moving objects whose shape changes considerably and overlap each other.

References

[1] A. Elgammal and L. S. Daivis, "M2Tracker: A Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene Using Region-Based Stereo", *Proc. European Conference Computer Vision*, 2002

[2] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale and S. Shafer, "Multi-camera Multi-person Tracking for EasyLiving", *Proc. 2nd IEEE Workshop on Visual Surveillance,* 1999

[3] G. Stein, "Tracking from Multiple View Points: Self-calibration of Space and Time", *Proc. IEEE CVPR Conference*, pp.521-527, 1999

[4] J. Kang, I. Cohen and G. Medioni, "Continuous Tracking Within and Across Camera Streams", *Proc. IEEE*

Conference on Computer Vision and Pattern Recognition, 2003

[5] I. Haritaoglu, D. Harwood, and L.S. Davis, "W4: Real-Time Surveillance of People and Their Actives", *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 809-830, Aug. 2000.

[6] N. Amamoto and A. Fujii, "De tecting Obstructions and Tracking Moving Objects by Image Processing Technique", *Electronics and Coom. Japan, Part 3*, vol. 82, no. 11, pp. 28-37, 1999.

[7] N. Ohta, "A Statistical Approach to Background Suppression for Surveillance System", *Proc. IEEE International Conference on Computer Vision*, pp. 481-486, 2001.

[8] G. L. Foresti, P. Moretti, V. Murino, G. Pettirossi, C. S. Regazzoni, "Hough-Based Extraction and Grouping of Symbolic Features", *Proc. 5th Workshop (Prometheus)*, pp. 109-118, Oct. 1991

[9] F. Oberti and C. Regazzoni, "Adaptive Tracking of Multiple Non Rigid Objects in Cluttered Scenes", *Proc. International Conference on Pattern Recognition,* 2000

[10] I. Cohen and G. Medioni, "Detecting and Tracking Moving Objects for Video Surveillance", *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 319-325, 1999

[11] I. Haritaoglu, D. Harwood and L. Davis, "W^4: Who? When? Where? What? A Real Time System for Detecting and Tracking People", *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 222-227, 1998

[12] Y. W. Wang, J. F. Doherty, R.Van Dyck, "Moving Object Tracking in Video", *Proc. IEEE 29th Applied Image Pattern Recognition Workshop,* 2000