RECORDING THE REGION OF INTEREST FROM FLYCAM PANORAMIC VIDEO

Xinding Sun*, Jonathan Foote[+], Don Kimber[+], B. S. Manjunath*

*Department of Electrical and Computer
Engineering, University of California
Santa Barbara, CA 93106
{xdsun, manj}@iplab.ece.ucsb.edu

[+]FX Palo Alto Laboratory, Inc.
3400 Hillview Avenue,
Palo Alto, CA 94304
{foote, kimber}@pal.xerox.com

ABSTRACT

A novel method for region of interest tracking and recording video is presented. The proposed method is based on the FlyCam system [4], which produces high resolution and wide-angle video sequences by stitching the video frames from multiple stationary cameras. The method integrates tracking and recording processes, and targets applications such as classroom lectures and video conferencing. First, the region of interest (which typically covers the speaker) is tracked using a Kalman filter. Then, the Kalman filter estimation results are used for virtual camera control and to record the video. The system has no physical camera motion and the virtual camera parameters are readily available for video indexing. The proposed system has been implemented for real time recording of lectures and presentations.

1. INTRODUCTION

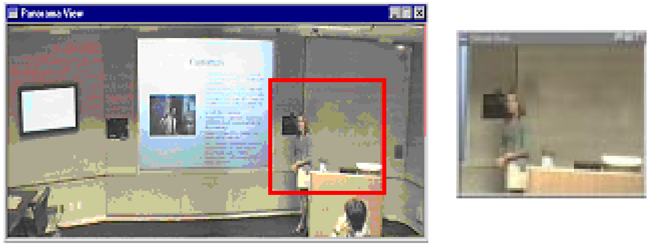
In recent years there has been a significant interest in research on camera-based applications such as automatic speaker capture in a seminar, class-room, or in a teleconference. A typical scenario is cameras automatically recording a speaker moving in front of a room. Figure 1 shows one such example. For wide-angle or panoramic cameras, it is often desirable to record only a small region of interest (ROI) of the larger image. Ideally, the ROI should contain the speaker's image, and that ROI movement should be smooth and comparable to that of cameras controlled by a human operator.

A practical solution to the above problem should take into consideration factors such as robustness, quality of the output video, feasibility of real-time computations, and overall cost. In the following sections, a low-cost system architecture using FlyCam [4] will be outlined and the tracking and recording problems will be addressed.

There are some commercial as well as research systems that attempt to provide a solution to this problem. Sony's EVI-D30 camera [8] can be used to track moving objects but it is often not robust. In particular, steerable cameras suffer from the drawback that tracking fails when the subject leaves the camera's field of view. Wang and Brandstein offer one solution [10] by combining microphone arrays and cameras for face tracking in a classroom/seminar setting. Systems that stitch multi-camera video sequences have been designed to capture events. For example, Majumder et al. [5], and Nicolescu and Medioni [6] have developed such systems to track objects in high-resolution video using virtual camera control. The FlyCam system [4] has similar functionalities to [5] and [6], but it is more compact and far less expensive, and therefore is a good match to the task here.

Previous person tracking efforts date back to the early 1980s. An example is Badler and O'Rourke's [7] work on 2-D kinematic modeling. In more recent work, Darrell et al. [3] integrates stereo, color, and face detection with person tracking. Since the main objective is tracking, the output of these systems is usually an object outline. Using raw tracking results to steer ROI selection usually produces objectionable jitter in the video output. In terms of recording processes, the most closely related work is by Stiefelhagen et al. [9]. In their work, a fixed panoramic camera is used to capture the scene around a table. However, their objective is to capture actions of participants to get the focus of attention, not to track and record them.

This paper presents a novel method that integrates tracking and recording. The system architecture is based on FlyCam. Multiple cameras are aligned to capture events in the front of a seminar room. Stitching the video frames from the multiple fixed cameras produces a high resolution and wide-angle video sequence. The ROI is then tracked using Kalman filtering from the object motion. The Kalman filter output is used to steer a "virtual camera" for recording video. The Kalman filter output is smoothed to simulate the response of a human camera operator (which will be discussed in the later sections). To our knowledge, such an approach has not been investigated before.



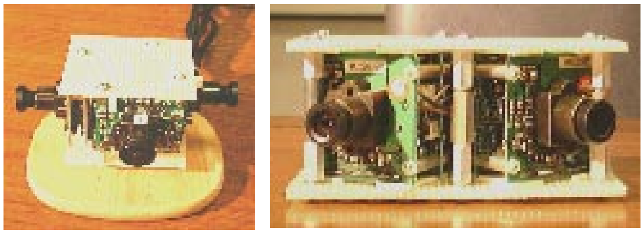
(a) 180º scene view (b) ROI output

Figure 1. An example of a panoramic scene and its ROI

2. SYSTEM ARCHITECTURE

2.1. The FlyCam System

Figure 2a shows the FlyCam system developed at FX Palo Alto Laboratory. This system generates panoramic video from multiple adjacent cameras in real time. Lens distortions are corrected and the images are stitched seamlessly by digital warping. Figure 2b shows a modified version of the panoramic one. It covers an 180º frontal view. A FlyCam is compact and inexpensive, using component cameras that cost around $100 each. The system presented here uses the 180º view FlyCam as the video input device.



(a) 360º view FlyCam (b) 180º view FlyCam

Figure 2. FlyCam examples.

2.2. General System Architecture

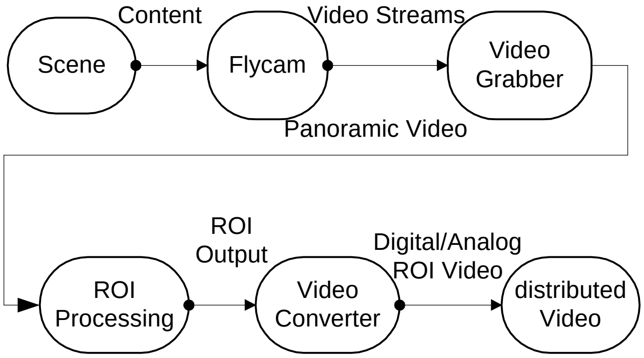


Figure 3. General System Architecture

Figure 3 shows the general structure of the proposed person tracking and recording system using an 180º view FlyCam. Each FlyCam component camera produces NTSC video that is digitized using a frame grabber. After stitching and ROI processing, the output digital video can be recorded or distributed, for example over the web. Additionally, the ROI video can be converted back to an analog signal for recording or distribution. After image stitching, the core part of the tracking system is the ROI processing, which is discussed in the following sections.

2.3. Tracking and Recording Components

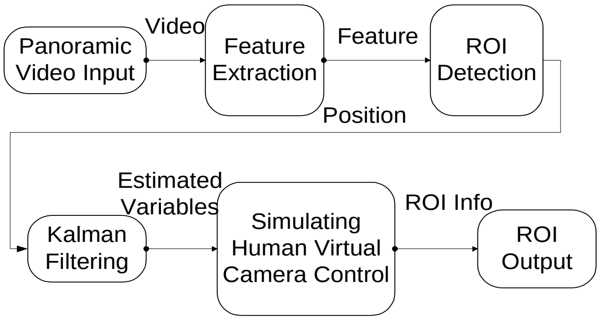


Figure 4. Integrated ROI Tracking and Recording.

The ROI processing consists of two parts: tracking and recording. The input to the tracking component is the stitched high-resolution wide-angle video from the FlyCam. In the tracking phase, the position of the ROI is detected from computed visual features. The detected position is then fed into a Kalman filter for position tracking. Estimation results from the output of the tracking process are then smoothed to produce the ROI output video. Figure 4 shows a general schematic of this ROI tracking and recording.

3. TRACKING

Many methods have been proposed in the literature for object tracking. Since the primary objective is to capture a single speaker in a panorama, complex models such as those used in [3] and [7] are not needed. The speaker is modeled as a point object corresponding to the centroid of the body. The ROI output is a predetermined rectangular region that surrounds this point. Thus, the ROI tracking basically tracks the body centroid.

3.1. Feature Extraction

Two principal features are considered for tracking: normal flow and color. The proposed solution is based on the overall confidence of motion and color change at each pixel. The confidence value is computed as a weighted sum of the color and motion information.

Given two consecutive frames, the standard optical flow equation based on pixel intensity is given by

$$I_xU + I_yV + I_t = 0 \tag{1}$$

Where I_x and I_y are the spatial derivative of the intensity in the x- and y- directions, respectively, U and V are the corresponding velocities, and I_t is the temporal derivative. The normal flow is typically defined ([1]) as:

$$V_n = \frac{I_t}{\sqrt{I_x^2 + I_y^2}} \tag{2}$$

This value is then normalized to [0,1], and is taken as the confidence of motion at the pixel. The normalized values are denoted by $C_m(x,y)$.

In addition to motion, color provides important information about the scene. While any of the traditional color spaces (such as RGB, HSV, Luv, etc.) can be used for the computations, it is observed that the HSV space is better suited for computing the color changes. Separating hue from the saturation and brightness adds robustness under most lighting variations. Bradski [2] uses the distribution of the hue (H) value for tracking. The pixel-wise hue difference between two consecutive frames is computed normalized to 0 to 1. The normalized value is defined as color confidence and denoted by $C_c(x,y)$.

The overall confidence value of motion and color change at each pixel can then be computed as a weighted sum of $C_c(x,y)$ and $C_m(x,y)$:

$$C(x,y) = w_1C_m(x,y) + w_2C_c(x,y) \tag{3}$$

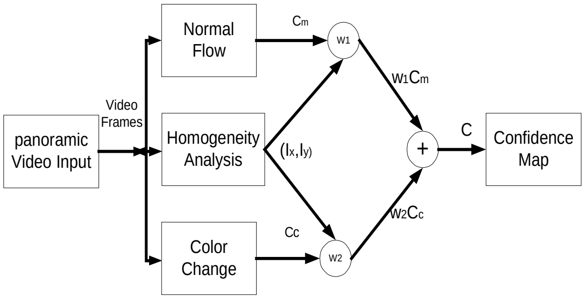


Figure 5. Building the Confidence Map

The weights in (3) can be fixed for simplicity. However, a better way to combine the motion and color information is to use a spatially varying weight according to the homogeneity of the image. This can be obtained directly from the spatial derivatives of the image as show in Figure 5. Dynamically combining the motion and color information limits the aperture problem inherited from the motion estimates. For example, if the spatial derivatives at a given pixel are very small, (2) tends to create large errors for normal flow estimation. In this case w_2 can be set to zero. After the confidence value at each pixel is computed, a confidence map for a given video frame is obtained. This confidence map is then used for feature tracking.

3.2. Detecting the Centroid

During the initialization process, thresholding the confidence map distinguishes the foreground from the scene background. The centroid of the body can be estimated using the first order spatial moment of the foreground, which yields a reasonable estimate of the speaker's centroid. After initialization, the system automatically tracks the centroid of the speaker as it moves.

Two centroid tracking methods were used. The first is essentially the same as the initialization process. The centroid of the foreground is computed for each video frame. The second method involves motion estimation. After initialization, the global flow of the ROI is computed using the confidence map with an affine model. The centroid information is updated from only the centroid flow. The first approach is faster though less smooth than the second. Because ultimately the Kalman filter is used to smooth the centroid estimate, the first method was chosen for its speed.

3.3. Tracking using a Kalman Filter

As mentioned, the estimate of the centroid coordinates is somewhat noisy. If the noise is assumed to be Gaussian, then it can be ameliorated using an extended Kalman filter.

The centroid has a track in the 2D space. The track in the x-direction can be modeled by the second-order Taylor series expansion of the form:

$$x(k+1) = x(k) + v_x(k) + a_x(k)T^2/2 + h.o.t. \tag{4}$$

$$v_x(k) = a_x(k)T + h.o.t. \tag{5}$$

Where $x(k)$ is the centroid coordinate in the x- direction, $v_x(k)$ is the corresponding velocity, $a_x(k)$ is the corresponding acceleration, and T is the time interval. Similarly, the same model applies to the track in the y- direction. Combining the models in two directions gives the centroid system model:

$$\mathbf{F}(k+1) = \Phi\mathbf{F}(k) + \Gamma\mathbf{w}(k) \tag{6}$$

Where $\mathbf{F}(k) = \left[x(k), y(k), v_x(k), v_y(k)\right]^t$ and $y(k)$ is the centroid coordinate in y- direction, and $v_y(k)$ is the corresponding velocity, while $\mathbf{w}(k)$ is the system Gaussian noise, representing the acceleration of the centroid in the x- and y- directions, and

$$\Phi = \begin{bmatrix} 1 & 0 & \mathrm{T} & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ \mathrm{T} & 0 \\ 0 & \mathrm{T} \end{bmatrix}$$

Higher order Taylor series expansion can be applied to the centroid system model, which leads to higher order of the system model. But experiments showed that this model worked well. In addition, as discussed in the later section, the system variables provide enough information for virtual camera control.

Since the speaker is modeled as a simple point, the measurement of it can be simply modeled as:

$$\mathbf{Z}(k) = H\mathbf{F}(k) + \mathbf{n}(k) \tag{7}$$

Where $\mathbf{Z}(k)$ is the measurement, $\mathbf{n}(k)$ is the measurement Gaussian noise and H is the measurement transfer function, in this case a scaling factor.

The covariance form of Kalman filtering is used to recursively update the prediction based on the innovation information at each step. The prediction at each update is output for further ROI recording purposes. The predicted or estimated variable used to control the recording process is $\hat{\mathbf{F}}(k) = [\hat{x}(k), \hat{y}(k), \hat{v}_x(k), \hat{v}_y(k)]^t$ which corresponds to the variable $\mathbf{F}(k)$.

4. RECORDING

Kalman filtering reduces most of the noise inherent in the tracking estimate, and suffices for most purposes. However, if the tracking result moves the recording ROI window directly, the quality of the output video is often jittery. The resulting motion is less smooth than that of a physical camera, which has inertia. Therefore, an additional filtering step is taken to produce smooth and pleasant ROI video output.

4.1. Virtual Camera Control

The method proposed here for virtual camera recording control is based on the following observation. When an experienced camera operator records the lecture, if the speaker is motionless or moving only in a small region, the operator usually does not move the camera (stabilization control). When the speaker changes his position by a large distance, the operator must move the camera to catch up with the speaker (transition control). After the speaker has been centered, the operator follows further movement (following control). Accordingly, the virtual camera control operates in three similar regimes.

Stabilization control is based on the Kalman filter estimates of position and velocity. The initial centroid position is registered first, denoted as $\mathbf{Y}_R(k) = [x_R(k), x_R(k)]^t$, where

$x_R(k), y_R(k)$ correspond to its coordinates at x- direction and y-direction respectively. Then at each frame, the estimated speed and position are checked. They can be obtained from $\hat{\mathbf{F}}(k)$ during the Kalman filter update process. If the following two conditions are satisfied, the virtual camera is fixed and the registered position is used as position output. First, the new position is within a specified distance of the registered position at a given direction. Second, the estimated speed is below a specified threshold at a given direction. Otherwise, the virtual camera control is changed to the transition regime. The stabilization control conditions can be formalized as:

$$\mathbf{Y}(k) = \mathbf{Y}_R(k) \qquad (8)$$

$$if \quad |\hat{x}(k) - x_R(k)| < \sigma_1,\ |\hat{y}(k) - y_R(k)| < \sigma_2$$

$$and \quad |\hat{v}_x(k)| < \sigma_3,\ |\hat{v}_y(k)| < \sigma_4$$

Where $\sigma_1, \sigma_2, \sigma_3,\ and\ \sigma_4$ are thresholds, $\mathbf{Y}(k)$ is the ROI output.

In the transition regime, a lowpass filter is used to update the virtual camera location. For this purpose, a first order low pass IIR filter is used:

$$\mathbf{Y}(k+1) = \alpha_1 \mathbf{Y}(k) + \alpha_2 \hat{\mathbf{X}}(k) \qquad (9)$$

Where $\alpha_1 + \alpha_2 = 1$, $\alpha_1, \alpha_2 > 0$, and $\hat{\mathbf{X}}(k) = [\hat{x}(k), \hat{y}(k)]^T$ is the estimated centroid from the Kalman filter and serves as the input of the IIR filter. The virtual camera now follows $\mathbf{Y}(k)$, which is smoother than the Kalman filter output. Experimentation showed that values of $\alpha_1 = 0.8, \alpha_2 = 0.2$ gave a reasonable simulation of human camera operation.

Since the IIR filter (9) tends to create delay in the output, the number of steps of in the virtual camera transition is limited. After a certain time in the transition regime, for example 0.5 second, the camera control is switched to the “following” regime. Updating the ROI position directly from Kalman filter output realizes this objective:

$$\mathbf{Y}(k) = \hat{\mathbf{X}}(k) \qquad (10)$$

Note that this is equal to setting $\alpha_1 = 0, \alpha_2 = 1,$ in the IIR filter (9).

4.2. Practical Considerations

The Kalman filter assumes environmental noise is Gaussian. This may not be the case for the application here. For example, the projection display and the audience both can produce constant noise in fixed regions in the background, as can be seen in Figure 1. This knowledge can be incorporated into the tracking system to improve performance, especially as the panoramic video cameras are fixed with respect to the background. Configuration parameters allow the confidence value to be changed or ignored in specified regions. For example, the region of the projection screen is weighted less so that slide changes or projected videos are not tracked in preference to the speaker. By offering this kind of flexibility, the tracking technology can be easily adapted to different environments.

5. CONCLUSION

In this paper a new method is presented for recording the region of interest in a wide-angle video. The method integrates tracking and recording processes, and simulates human camera control for recording the speaker during lectures and presentations. The whole process is automatic, robust and fast enough for real time applications. The method is designed for 180º view events, however, it can be applied to 360º view as well, provided there is only one moving speaker present in the scene. The system has been implemented in the FXPAL “Kumo” conference room for recording seminars and presentations.

For typical lectures, the speaker remains at roughly the same distance from the camera, thus zooming is not necessary. However, zooming could be achieved by scaling the ROI for applications that needed it, as discussed in [9] and [10].

Since in the system the cameras are stationary, the tracking information also provides a feature description of the video content, which is useful for MPEG7 related applications. Also, since the region of interest is isolated from other objects in the scene, the recording result may be useful for MPEG4 object based coding. Other research possibilities include analyzing speaker activity, or using the ROI image as the basis for gesture tracking or face recognition.

6. REFERENCES

[1] J. L. Barron, D. J. Fleet and S. S. Beauchemin, "Systems and Experiment Performance of Optical Flow Techniques,” *Intern. J. Comput. Vis*, 12:1, pp. 43-77, 1994.

[2] G. R. Bradski, “Real time face and object tracking as a component of a perceptual user interface,” *Proc. WACV’98*, pp. 214-19,1998.

[3] T. Darrell, G. Gordon, M. Harville, and J. Woodfill, “Integrated person tracking using stereo, color, and pattern detection,” *Proc. CVPR’98*, pp. 601-608, 1998.

[4] J. Foote, and D. Kimber, “FlyCam: practical panoramic video and automatic camera control,” *Proc. ICME’2000*, pp. 1419-1422, 2000.

[5] A. Majumder, W. B. Seales, M. Gopi, and H. Fuchs, “ Immersive teleconferencing: a new algorithm to generate seamless panoramic video imagery,” *Proc ACM Multimedia’99*, pp.169-178, 1999.

[6] M. Nicolescu and G. Medioni, “Electronic pan-tilt-zoom: a solution for intelligent room systems, ” *Proc. ICME’2000*, pp. 1581-1584, 2000.

[7] J. O'Rourke, N.I. Badler. “Model-based image analysis of human motion using constraint propagation,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2(6), pp. 522-536, 1980.

[8] Sony EVI-D30, www.sony.com.

[9] R. Stiefelhagen, J. Yang, and A. Waibel, “ Modeling Focus of Attention for Meeting Indexing, “ *ACM multimedia’99* pp. 3-10, 1999.

[10] C. Wang and M. S. Brandstein , “A Hybrid Real-Time Face Tracking System," *Proc. ICASSP’98*, pp. 3737-3740, 1998.