

Automatic Cameraman with Trajectory Extraction

Jaime García Giráldez, Jose Luis Lamas, Yuri López de Meneses, Jacques Jacot

Laboratoire de Production Microtechnique (LPM-IPR) Ecole Polytechnique Fédérale de Lausanne

1015 Lausanne, Switzerland

`jaime.garciagiralddez@epfl.ch`

1 Introduction

Target tracking systems are a typical problem in Computer Vision. In this paper a vision-based automated cameraman is described. Several systems based on mobile cameras have been proposed [1, 2, 3, 4] with the goal to track a target and to maintain its position in the middle of the image. These systems are limited to a single object tracking and they do not extract any information about the trajectories. Our approach to the cameraman system is to increase the tracking robustness and to allow the user to extract more details of the tracked object.

The system is based on two cameras, a fixed and a mobile cameras. The fixed camera is pointed to the scene where the objects have to be tracked. The images from this camera are typically composed of a static background, so we have used a background subtraction technique to segmentation [5, 6, 7]. The position of the tracked object is used to drive the mobile camera and to point it to the object.

The advantage of using a fixed camera is that the trajectory of the object can be extracted. With this trajectory a more accurate drive of the mobile camera can be made, because we can predict its future position. The full system becomes a cornerstone on which higher-level functions can be built: face recognition with the mobile camera image or trajectory analysis and multiple object tracking with the fixed camera image.

2 Fixed Camera

The first stage is to obtain the position and the trajectory of the target. It is necessary to extract its position from the image and to transform it to world coordinates. In our system we assume that the tracked objects are moving in a plane, so this transformation will be possible with only one cam-

era (section 2.3).

The objective of the tracking is to give the position of the object as quick as possible, because of the movement of this object. This movement causes an error between the extracted position and the actual position, due to the time for image processing. Therefore, a fast image processing will be necessary to have an accurate mobile camera motion. Our system is able to work up to 25 images/second. However, a prediction of the position will be done to minimize this error.

The tracking system is based on a background subtraction using the color information. We use a statistical learning to model the background. Finally, with the camera position calculated by a camera calibration, we can transform the object position in the image to a real position. To reduce the processing time, we can use a region of interest (ROI) of the whole image. This ROI is defined by a predicted position of the object, which is obtained using a Kalman filter. Figure 1 shows the functional block diagram of the system.

2.1 Background Learning

Our objective is to build a color background model. To this end we build a statistic of the intensity value of each pixel over time. If a pixel variance remains within a threshold calculated by this statistic, we can update the color background using the mean color of this pixel. With this method we can reduce the random noise and the small movements in the scene. The background is updated in every new frame to reduce the effects of small changes of illumination or possible movements of the objects that make up the background.

2.2 Segmentation

Once the color background model is made, it is possible to extract the moving objects. To extract

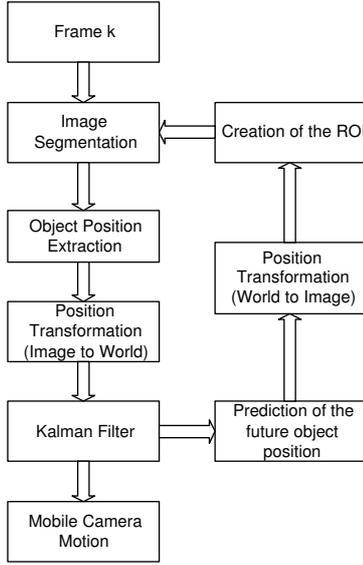


Figure 1: System functional block diagram

the objects the system uses the chromaticity coordinates r, g and b of the background model and the actual image[5]:

$$r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}, b = \frac{B}{R+G+B}$$

where $r+g+b = 1$.

Using the chromaticity coordinates in detection has the advantage of being more insensitive to small changes in illumination. We reduce the shadow effects too; this is important because the shadows can modify the desired object position.

Since we have the object blob defined by the pixels that do not belong to the background, we can obtain its position with the projection of the center of gravity on the lowest part of the blob.

2.3 Camera Calibration and Coordinate Transformation

In the previous section it was shown how to find the object position in the image. Now the system needs to transform this image position to a real position. Therefore, a camera model has to be used[8][9]. Tsai's camera model [10] can calculate up to 11 camera parameters. However, we do not need to know all these parameters; we only need to transform a pixel position to a specific plane position where the objects are moving. So we use a reduced model based on the pin-hole model that

was developed at the LPM[11]. This camera model needs only 4 points of this plane to be calibrated (equation 2.3) while Tsai's model needs 11.

$$\begin{bmatrix} X_{image} \\ Y_{image} \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} X_{real} \\ Y_{real} \\ 1 \end{bmatrix}$$

where: X_{image} and Y_{image} image coordinates
 X_{real} and Y_{real} world coordinates
 m_{ij} transformation matrix coefficients

2.4 Kalman filter

The Kalman filter [12] is a set of mathematical equations that provides an efficient computational solution of the least-squares method. The filter is very useful for tracking systems because it support estimations of past, present and even future states, and it can do so even when the precise nature of the modelled system is unknown. To make our system fast and accurate, we use the $\alpha - \beta$ Kalman Filter, which is a first order approximation of the Kalman process [13].

The noise due to a bad segmentation in the tracking stage is minimized by the Kalman filter. Moreover, it gives the trajectory and the behavior of the object. This trajectory and the prediction of the next position are used later for the mobile camera motion.

3 Automatic Mobile Camera

The next stage is to drive the mobile camera. The objective is to have the tracked object in the center of the mobile camera image. For this purpose a miniaturized pan-tilt camera has been used. To drive this camera it is necessary to transform the position of the object tracked to a pan-tilt position (figure 2), so a camera calibration is needed.

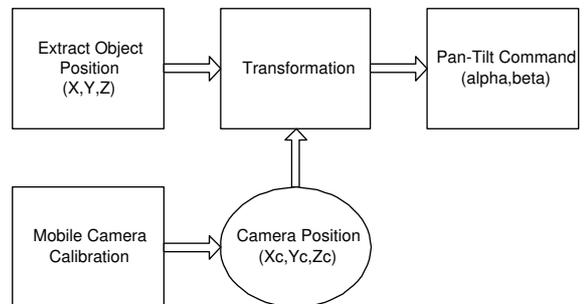


Figure 2: Mobile Camera Motion Algorithm

3.1 Mobile Camera Calibration

There are 5 unknown parameters for the mobile camera:

- 3D Mobile Camera Position (X_c, Y_c, Z_c)
- Initial Pan-Tilt Offsets (α_0, β_0)

The camera position is necessary to transform the object position to pan-tilt angles. To have a better precision, the calibration computes the initial offsets of the home position.

We have made a calibration that needs only 4 points of the plane and their corresponding pan-tilt positions (figure 3). These 4 points are the same as in the fixed camera calibration, because we need to have the same reference system for the object position and the mobile camera position.

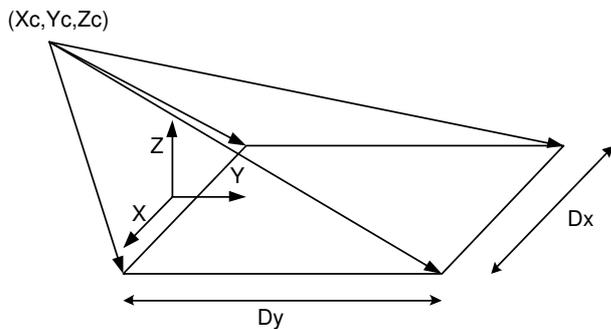


Figure 3: Mobile Camera Calibration

The camera position will be the intersection of four lines. Every line is made of point in the plane and a vector to the camera that can be expressed by angles α and β :

$$(X, Y, Z) = (X_i, Y_i, Z_i) + K \cdot \vec{V}_i$$

$$\vec{V}_i = (\cos(\alpha_i)\sin(\beta_i), \sin(\alpha_i)\sin(\beta_i), \cos(\beta_i))$$

This intersection and the possible offsets are computed by a recursive least square method.

3.2 Mobile Camera Motion

Since the mobile camera position is computed, it is possible to find the pan-tilt coordinates (α and β) to point the camera to the object:

$$\alpha_i = \arctan\left(\frac{Y_i - Y_c}{X_i - X_c}\right)$$

$$\beta_i = \arccos\left(\frac{Z_i - Z_c}{M}\right)$$

$$\text{with } M = \sqrt{(X_i - X_c)^2 + (Y_i - Y_c)^2 + (Z_i - Z_c)^2}.$$

Due to the image processing delay (40 msec) and the time of mobile camera control (approx. 20 msec) a prediction of the position of the object has been used to compute these pan-tilt coordinates. This prediction uses the position of the object and its speed measured by the Kalman filter (section 2.4).

4 Tests and Results

The system has been tested in several environments. The use of this system is restricted to environments with a static background. Figures 4 and 5 are some examples of the object extraction stage in a typical video surveillance scene.

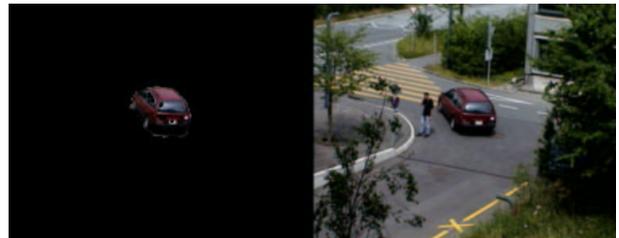


Figure 4: Image Segmentation



Figure 5: Image Segmentation

In figure 6 the full system is shown: trajectory extraction in the left side, the images from the fixed cameras and the images from the mobile cameras. This is a test which tries to analyze the behavior of a tennisman; with these images it is possible to analyze the technique (mobile camera image) and strategy of the player (trajectory).

In the case of video surveillance, the automatic mobile camera allows the user to have a higher level of security, because it helps to recognize the objects more easily. A zoom can be added to have a view of the object.

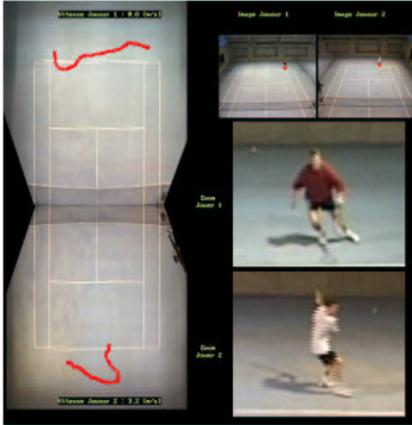


Figure 6: Images and Trajectory Representation

The robustness of the system was tested with a tennisman. A tennisman is characterized by a high velocity of movements and a very fast changes of direction. These changes of direction are a critical stage for mobile systems. Moreover, there is also the error in the segmentation stage. The tests has done a segmentation mean error of 15 pixels, which is equivalent to 1 meter of error of position with 20 meters between the camera and the tracked object. However, the movement and the segmentation problems are minimized due to the fast image processing (25 images/sec) and the prediction of position (Kalman Filter).

5 Conclusion

In this paper we have presented our approach for a tracking system. The system joins the advantages of the fixed camera systems (moving objects and trajectory extraction) and the mobile camera systems (video surveillance, automatic cameraman). Moreover, the system is a good base for a new higher-level systems, like behavior and trajectory analysis or face recognition.

References

- [1] D. W. C. David C. Woo, "3D visual tracking using a network of low-cost pan/tilt cameras," Department of Electrical and Computer Engineering, McMaster University, Hamilton.
- [2] J.-L. F. S. T. Raphael Canals, Anthony Rousel, "A biprocessor-oriented vision-based tar-

get tracking system," *IEEE Transactions on Industrial Electronics*, vol. 49, April 2002.

- [3] J. P. C. Luis Jordão, Matteo Perrone, "Active face and feature tracking," Instituto de Sistemas e Robótica - Instituto Superior Técnico, Portugal.
- [4] G. L. Swarup Reddi, "Analysis of camera behavior during tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, August 1995.
- [5] L. D. Ahmed Elgammal, David Harwood, "Non-parametric model for background subtraction," *Proceedings of IEEE ICCV99 FRAME-RATE Workshop*, 1999.
- [6] A. C. Francesco Ziliani, "Image analysis for video surveillance based on spatial regularization of a statistical model-based change detection," *Real-Time Imaging*, 2001.
- [7] A. Makarov, "Comparison of background extraction based intrusion detection algorithms," *Proc. IEEE International Conference on Image Processing*, pp. 521–524, September 1996.
- [8] O. Faugeras, *Three-Dimensional Computer Vision*. MIT Press, 3rd ed., 1999.
- [9] A. Z. Richard Hartley, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [10] R. Y. Tsai, "An efficient and accurate camera calibration technique for 3d machine vision," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 364–374, Miami Beach 1986.
- [11] C. Wille, "Tracking des lignes d'un court de tennis pour l'estimation de la matrice de projection d'une caméra PTZ," tech. rep., Groupe Vision IPM-EPFL, 2001.
- [12] G. B. Greg Welch, "An introduction to the kalman filter," Tech. Rep. 95-041, Department of Computer Science, University of North Carolina at Chapel Hill, 1998.
- [13] T. C. P.L. Rawicz, P.R. Kalata and K. Murphy, "On $\alpha-\beta$ target tracking initiation," *Proceedings of the American Control Conference, Philadelphia*, June 1998.