A Computer Vision Based Camera Pedestal's Vertical Motion Control

Richard Y. D. Xu [1], Joshua M. Brown [2], Jason M. Traish [3], Daniel Dezwa[4]
Charles Sturt University, NSW, Australia
rxu[1],jtraish[3]@csu.edu.au
jbrown86[2],ddezwa01[4]@postoffice.csu.edu.au

Abstract

.

Traditional camera pedestals are manually operated. Our long term goal is to construct a fully autonomous pedestal system which can respond to changes in a scene and mimicking the human camera operator. In this paper, we discuss our experiments to control the vertical motion of a pedestal by leveling its position with a human head or a tracked hand-held object. We describe a set of computer vision methods used in these experiments, including the head position tracking using Gaussian Mixture Model (GMM) of the foreground blob and hand-held object tracking using Continuously Adaptive Mean shift (CAM-shift) with motion initialization. We also discuss the application of Kalman Filter and showing its effect in the reduction of the number of jittering pedestal motions.

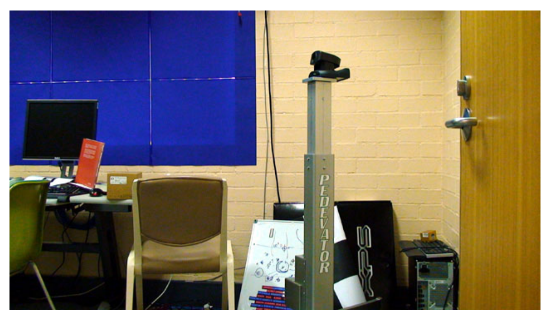
1. Introduction

Robotic pedestal is a term commonly refers to in the TV industry, as a manually-operated camera pan-tilt (PT) head mounted on a vertical pedestal. There are several commercial manufacturers of robotic pedestals available, including Vinten [1] and Hitachi [2]. Most recently, some robotic pedestals (such as Vinten Radamec [1]) are equipped with a planar motion base and an interface software for pre-programmed movements.

At the same time, there has been some work in the area of automated Pan-Tilt-Zoom (PTZ) camera controls. The typical applications of these research are found in automated instructional video capture [3, 4] and video surveillance and analysis [5, 6].

In this paper, we will describe two experiments, which we use to control the vertical motions of a robotic pedestal and the rest of this paper is organized as follows: In section 2, we describe the hardware used and in section 3, we describe the computer vision techniques that we use to determine the 3D position of a human head and a coloured hand-held object. In section 4, we present our experimental results, in particularly, the results obtained after Kalman Filter smoothing, which helps to remove most of the jittery vertical motions.

(a)



(b)

Fig. 1. The hardware used in our project: (a) a pair of static cameras, PIKE032 and the processing PC (b) the Pedevator

2. Hardware

The robotic pedestal used in our experiment is Hitachi Eagle Pedevator. The 3D position determination (human head or the colour object) is achieved by tracking the corresponding image features from a pair of wall-mounted static cameras (AVT

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

PIKE 032). The communication with the Pedevator is using a set of RS232/RS485 commands from the PC.

3. Computer vision based vertical position inputs

We calibrate the static cameras (both their intrinsic and extrinsic parameters) using planar chessboard grids [7], which is a standard method used in most camera calibration applications. The 3D position of an object, in particular, its y-coordinate is determined by tracking the corresponding object's 2D position in both camera views.

3.1 Head position tracking

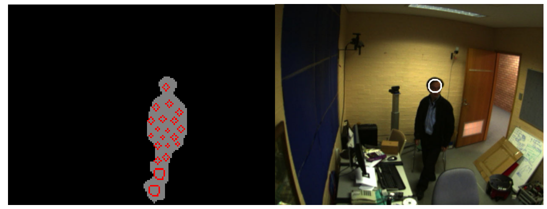
Since one of our long term research aims is for the robotic pedestal be able to "learn" from a human camera operator, by "*mimicking*" his/her operations. As a first step, we programmed the pedestal to move up and down depends on the human camera operator's centre of head position (which his eye position have close proximity of).

The method which we use to determine the 3D position of the person's centre of the head is described in our earlier paper [6]. In essence, we first calculate the moving foreground using background subtraction [8] technique. We then use the location data (2d feature) of every pixel belonging to the foreground blob, and to model them using Gaussian Mixture Model (GMM), where in order to improve the efficiency, we let $\sum = \sigma_k^2 I$, where I is a 2D identity matrix. We reject any cluster with σ_k greater than two times the median σ value of all clusters to combat the problem of noisy foreground due to noises and shadows. A detailed description of this approach is found in [6]. The results from a camera view are shown in figure 2.

A video demo showing our head position tracking is found in:

http://silica.csu.edu.au/staff/cs/rxu/videos/ped_background.wmv

(a) (b)



(c) (d)

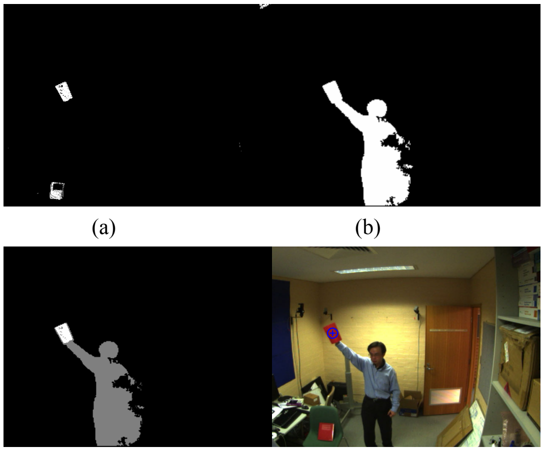
Fig. 2. Detecting the human head: (a) background subtraction result (b) classification of pixel locations in the foreground using GMM (c) the circles showing each elements of the mixture, where the centre of a circle specifies the mean value and the radius indicates the standard deviation of each mixture (d) the tracking result

3.2 Motion initialized colour histogram segmentation of hand-held object

Our second method for pedestal control is based on estimating the 3D position of a hand-held object of a special colour. In order to track object of colours in the two camera views, we used approaches based on Continuously Adaptive Mean-shift (CAM-Shift) [9] which is a robust nonparametric technique for climbing density gradients to find the mode of probability distributions using mean-shift algorithm.

In general, the problem of identifying objects based on colour feature is always affected by the presence of other objects with similar colour profile. Although CAM-Shift algorithm starts the search region on the current video frame based on the previously identified region of interest, however, the problem of locating the initial search window at *t0* is a problematic one. A global search may return multiple segmented regions, shown in figure 3.d, where the book and the hand-held object has similar colour histograms.

However, we have noticed that our object of interest is a hand-held one which in some stage, is in motion. Therefore, we can resolve our window initialization problem by refining the search region within the pixels belonging to the foreground blob initially. In our approach, a fast, adaptive background subtraction method is employed [10] which has short learning time (such that the objects moves will initially be classified as foreground but will soon blend into the background after a short period of time when it stays stationery). After a threshold of time, the region identified within the foreground blob will be used as the starting search window for the CAM-Shift algorithm. In the case of tracking object is lost, this motion-based initialization procedure resumes again.



(a) (b)

(c) (d)

Fig. 3. Motion initiated colour segmentation result: (a) the scene containing multiple regions, which some is falsely detected by colour histogram segmentation. (b) Foreground blob in white. (c) the colour histogram segmentation result (in white) on the foreground blob (grey) (d) the tracked object.

A video demo showing the hand-held object tracking is shown on:
http://silica.csu.edu.au/staff/cs/rxu/videos/ped_hand_track.wmv

4. Experiments with Kalman filtering

We have built a prototype system using methods described in each of the preceding sections. The camera images are streamed into the testing PC. A set of RS232/RS485 commands is then sent to the Pedevator via the serial ports based on the protocols provided by the manufacturer.

The prototype was coded in Visual C++ and we have used Intel OpenCV functionalities for some of the image processing and computer vision implementations.

Because Hitachi Pedevator has a limited moving range between 127 cm and 86 cm. We use a linear function to map the y-value of the tracked object $y_{tracking\_pos}$ to the predefined Pedevator setting ψ. In our project, we specify $y_{tracking\_\max}$ = 180cm and $y_{tracking\_\min}$ =10cm.

$$\psi = \lfloor y_{tracking\_pos} / (y_{tracking\_\max} - y_{tracking\_\min}) \rfloor \times 32$$

where $\lfloor \quad \rfloor$ is the floor function, and $\psi \in \{0..31\}$

In order to achieve this purpose, we have calibrated the Pedevator, such that a range of 32 predefined positions are defined which the Pedevator would move equal distance between each of the adjacent settings.

What we have noticed in the first instance is that there have been a lot of jittery movements in Pedevator. This is due to the noises in the 3D tracking. In order to combat this problem, we apply Kalman Filter for smoothing the trajectories of tracking subject.

The effects of Kalman filter smoothing is shown in figure 4, where the green lines are the 3D point tracking prior to apply Kalman filtering and the blue line is showing 3D point estimation after Kalman filter smoothing. This data is collected during the hand-held object tracking.

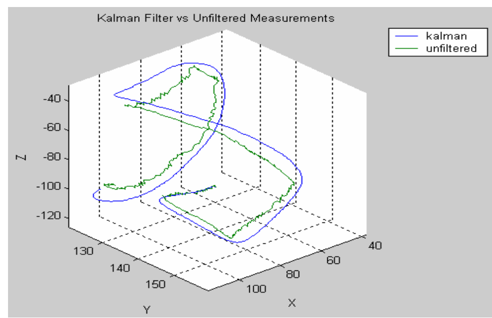
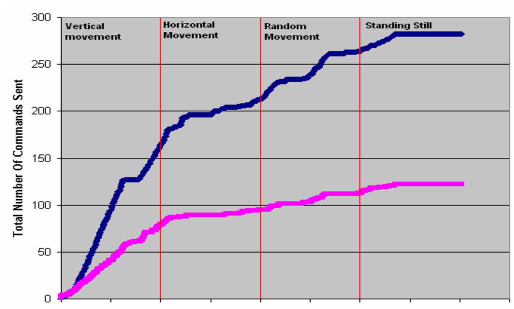
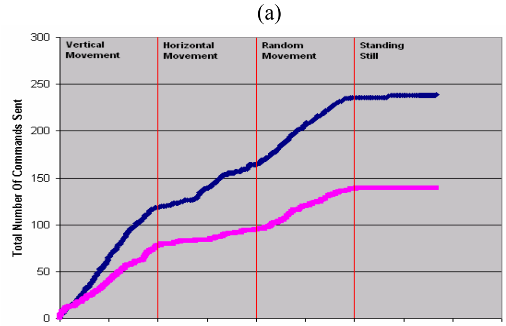


Fig. 4. The effect of Kalman filter smoothing, showing the trajectory prior (green line) and after (blue line) the application of Kalman Filtering



(a)



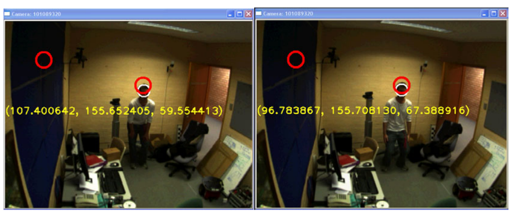
(b)

Fig. 5. The diagram showing before and after the application of Kalman Filter for (a) hand-held object tracking (a) head tracking, during the four stages that a person performs *vertical*, *horizontal*, *random* and *standing still*

We have also shown in figure 5, the number of movement control commands were sent to the Pedevator before and after Kalman filter were applied. As the number of controls sent to the Pedevator reduces, the jittery motions reduce significantly.

5. Results and Conclusion

In figure 6, we show the screen captures (from one of the two camera views) of the two tracking methods used. The full results have been uploaded to the following video link:



http://silica.csu.edu.au/staff/cs/rxu/videos/ped_background.wmv

(a)

http://silica.csu.edu.au/staff/cs/rxu/videos/ped_hand_track.wmv

(b)

Fig. 6. A sequence of images showing that the (a) human head and (b) hand-held object corresponds with the vertical movements of the Pedevator

In this paper, we have illustrated two user interaction methods which we have exploited to control the vertical motions of a robotic pedestal through RS232/RS485 protocol: The tracking by head position as well as tracking by hand-held object using motion initiated colour histogram segmentation. In both methods, a set of computer vision and pattern recognition techniques were used, including background subtraction, CAM-shift and Gaussian Mixture Modeling using EM algorithm.

We have shown that by applying Kalman filter smoothing, the number of jittery motions sent to Pedevator have reduced significantly.

6. Future work

The results achieved in this paper, will be joined by other studies, such as PTZ camera calibration [6], to achieve a fully autonomous robotic pedestals in the future. Some preliminary studies have also been carried out in the areas of CLR trajectory tracking using cubic curvature polynomials as well as studies into the efficient six-degree-of-freedom camera calibration methods, which extends the current PTZ camera calibration into vertical motion (y) as well as planar motion (x , z).

7. Acknowledgment

We thank the Dean, John Hicks for providing the summer research funding to the research students, which enables this project to complete on time.

References

[1] Vinten, "http://www.vintenradamec.com/fp145.php," 2006.
[2] Pedevator, "http://www.eaglepantilt.com/," 2006.
[3] A. Shimada, A. Suganuma, and R. Taniguchi, "Automatic Camera Control System for a Distant Lecture Based on Estimation of Teacher's Behavior," in *International Conference on Computers and Advanced Technology in Education*, 2004, pp. 428-158.
[4] M. Onishi and K. Fukunaga, "Shooting the lecture scene using computer-controlled cameras based on situation understanding and evaluation of video images," in *17th International Conference on Pattern Recognition*, Singapore, 2004, pp. 781-784.
[5] A. Hampapur, S. Pankanti, A. Senior, Y.-L. Tian, L. Brown, and R. Bolle, "Face cataloger: Multi-scale imaging for relating identity to location," in *IEEE Conference on Advanced Video and Signal Based Surveillance*, Miami, FL, 2003.
[6] R. Y. D. Xu, J. B. Gao, and M. Antolovich, "Novel Methods for High-Resolution Facial Image Capture using Calibrated PTZ and Static Cameras," in *IEEE International Conference on Multimedia & Expo (ICME 2008)*, Hannover, Germany, 2008, pp. 45-48.
[7] Z. Zhang, "Flexible Camera Calibration By Viewing a Plane From Unknown Orientations," in *International Conference on Computer Vision*, Corfu, Greece, 1999, pp. 666-673.
[8] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE CVPR*, 1999, pp. 246-252.
[9] G. R. Bradski, "Real Time Face and Object Tracking as a Component of a Perceptual User Interface," in *IEEE WACV 1998*, 1998, pp. 214-219.
[10] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian, "Foreground object detection from videos containing complex background," in *ACM international conference on Multimedia*, 2003, pp. 2 - 10.