151 S. Kang, J. Paik, A. Koschan, B. Abidi, and M. A. Abidi, "Real-time video tracking using PTZ cameras," Proc. of SPIE 6th International Conference on Quality Control by Artificial Vision, Vol. 5132, pp. 103-111, Gatlinburg, TN, May 2003.

Real-time video tracking using PTZ cameras

Sangkyu Kang*[a], Joonki Paik[ab], Andreas Koschan[a], Besma Abidi[a], and Mongi A. Abidi[a]

[a]IRIS Lab, Department of Electrical & Computer Engineering, University of Tennessee
Knoxville, Tennessee 37996-2100

[b]Image Processing Laboratory, Department of Image Engineering
Graduate School of Advanced Imaging Science, Multimedia, and Film,
Chung-Ang University, 221 Huksuk-Dong, Dongjak-Gu, Seoul, 157-756, Korea

ABSTRACT

Automatic tracking is essential for a 24 hours intruder-detection and, more generally, a surveillance system. This paper presents an adaptive background generation and the corresponding moving region detection techniques for a Pan-Tilt-Zoom (PTZ) camera using a geometric transform-based mosaicing method. A complete system including adaptive background generation, moving regions extraction and tracking is evaluated using realistic experimental results. More specifically, experimental results include generated background images, a moving region, and input video with bounding boxes around moving objects. This experiment shows that the proposed system can be used to monitor moving targets in widely open areas by automatic panning and tilting in real-time.

Keywords: Video tracking, adaptive background generation, object extraction

1. INTRODUCTION

Since September 11, the desire for safety in public locations, such as airports and government buildings where many people gather, has been increasing. For security purposes, these places usually adopt video-based surveillance that can record scenes by fixed or PTZ cameras, which can change their viewing area by using predefined rotation tables and timers. The recorded video can be examined often intrusions or accidents have occurred; however, this means that the system does not provide a real-time warning, which is very important to prevent accidents. For detecting an intruder or tracking suspects with current systems, the video surveillance system should be monitored by human operators, but this is not a simple task even for the skillful operators given the tedious task of watching video for more than a couple of hours. Also monitoring a large area requires many operators at the same time.

Recently, many tracking algorithms have been developed, and can be categorized into three groups: adaptive background generation and subtraction,[1,2] tracking using shape information,[3,4] and region based tracking.[5,6] Adaptive background generation is a very efficient way to extract moving objects, and the Gaussian distribution for each pixel is usually used to build a static gray background[1] or color background[7]. This method, however, requires stationary cameras to build background, and various research was proposed to extend the camera's view to track objects in large areas. L. Lee et al proposed a method to align the ground plane across multiple views to build a common coordinate for multiple cameras[8]. An omni-directional camera was also used to extend the field of view to 360°[9] with adaptive background generation and subtraction, but detected objects, such as a moving person or intruder, are usually at very low-resolution since one camera is used to grab the entire surrounding. Tracking using shape information is very interesting when used to track known objects or an object in the data base by B-spline contours or active shape models, but this method usually requires initial placement of an initial shape, which needs to be placed as near as possible to the targets, to get accurate tracking

* sangkyu@iristown.engr.utk.edu, phone: +1-865-974-9685, fax: +1-865-974-5459, http://imaging.utk.edu

results manually, and is usually not adequate for automatic tracking. Region based tracking also shows good tracking results, but these methods also require initialization steps on targets to build color distribution models or templates.

This paper describes an adaptive background generation algorithm for PTZ cameras using internal parameters of the PTZ cameras, such as pan/tilt angles. An overall system will be presented including background generation, moving region detection and extraction, and tracking as well as experimental results.

2. BACKGROUND MODELING AND TRACKING

2.1 Background modeling for a fixed view

Moving objects can be identified by the difference between the background and the current image. The simplest way is by using an image without any motion, but this method is not easy to be achieved when many people are present in a scene with illumination changes causing false positive alarms. The well-know method using Gaussian distribution can be found in Haritaoglu et al. [1] for gray scale images. This method computes the mean value over approximately 20-40 seconds, and the standard deviation of the median value is used to generate a background model. If the distance between an incoming pixel and the computed mean is larger than the standard deviation, the incoming pixel can be considered as a moving pixel, or background pixel when the distance is smaller than the standard deviation. This algorithm provided considerably improved results for extracting moving parts, but the long training time (20-40 seconds) for initialization is a disadvantage.

We use, as an alternative, a simple background generation method that requires a shorter training time. Figure 1 depicts this algorithm. In this method, a pixel that does not change for N consecutive frames is used to update the background image.

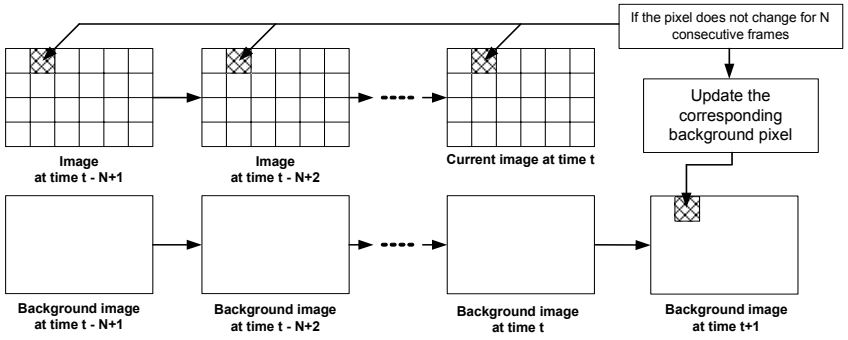


Figure 1: The proposed background generation algorithm.

The procedure implemented for background generation is summarized as follows.

1. If an image is the first image, set the first image as the initial background image, and set $Flag_{BG}(x,y)=0$.

2. Otherwise repeat the following steps for all pixels.

If $\left|I_k(x,y)-B_k(x,y)\right| < T_{BG}$: $I_k(x,y)$ is static.

if $(Flag_{BG}(x,y)>0)$, then $Flag_{BG}(x,y)=Flag_{BG}(x,y)-1$

else : $I_k(x,y)$ or $B_k(x,y)$ is not static.

$Flag_{BG}(x,y)=Flag_{BG}(x,y)+1$

if ($\left|I_k(x,y)-I_{k-1}(x,y)\right| < T_{BG}$ and $\left|I_k(x,y)-I_{k-2}(x,y)\right| < T_{BG}$) : $I_k(x,y)$ is static.

if ($Flag_{BG}(x,y) \geq N$), then $B_k(x,y)=I_k(x,y)$, $Flag_{BG}(x,y)=0$

else if $(Flag_{BG}(x,y)>0)$, then $Flag_{BG}(x,y)=Flag_{BG}(x,y)-1$

3. Repeat step 2 for every incoming image.

$I_k(x,y)$ and $B_k(x,y)$ represent the (x,y)-th pixel for the input image and the background image, respectively, at time $t=k$. N is the number of consecutive frames as shown in Figure 1, and $Flag_{BG}(x,y)$ is the current number of consecutive frames for the (x,y)-th pixel. The variable N typically depends on the frame rate of input sequences, and we used $N=5$ for 5 frames/sec video. All operations are performed in the gray-scale pixel intensity domain.

2.2 Background modeling for PTZ cameras

An adaptive background generation for stationary cameras and its application for PTZ cameras are shown in Figure 2. When a single, fixed camera is used, each location of a stationary pixel, denoted by the black rectangles in Figure 2(a), is time-invariant. Figure 2(b) shows that the location of the black rectangles changes depending on the camera's motion, that is, tilting and panning. The rectified images in terms of position of corresponding points are shown in Figure 2(c). Mosaicing enables a generated background for one view point to be reused as the background for the other view if there is an overlapped region. This overlapping condition holds in most practical situations because there will always be overlapped regions when a PTZ camera tracks objects.

Two methods can be used to register images with different pan and tilt angles; one is to use the 8-points algorithm to compute the homogeneous matrix and the other is to use the pan/tilt angle and a focal length. The 8-points algorithm requires at least 4 corresponding points from two images, and these points are used to compute the relation between images, which is a homogeneous matrix. If we can change the pan and tilt angle by the same amount, and once we compute the homogeneous matrix for this amount, for example 30˚ for the pan angle, 10˚ for the tilt angle at a zoom ratio of 1x, then we always get the same corresponding locations for a pair of images and the same homogeneous matrix. The first task for this method is to choose suitable pan and tilt angles for tracking. These angles should be selected to guarantee at least 50% overlapping depending on the zoom ratio since we need to track objects continuously. After selecting these angles, we can compute the homogeneous matrix, which describes a relation between a pair of images that have different pan and tilt angles. The computed homogeneous matrix is used to project the computed background into the new image that has the new pan/tilt angles. The advantage of this method is that we do not have to know the internal parameters of the PTZ camera, such as the size of the CCD sensor and the focal length, but we do need to compute homogeneous matrices for every possible combination of pan and tilt angles.

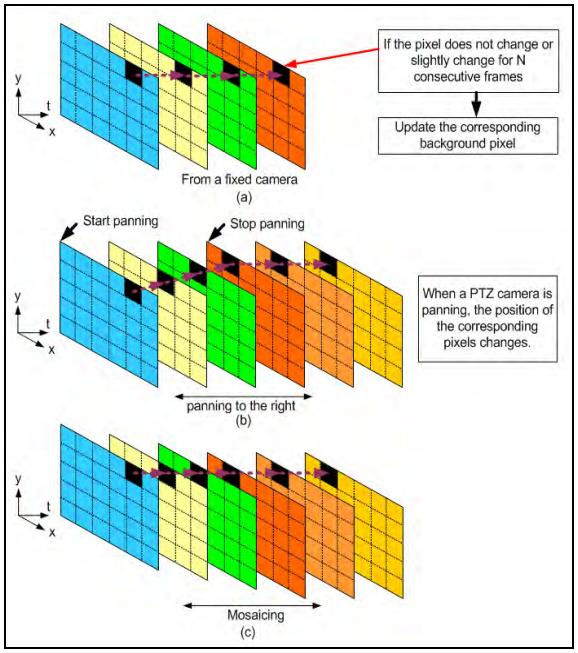


Figure 2: Adaptive background generation for stationary cameras and its application to PTZ cameras: (a) images captured by a fixed camera, (b) images captured by a panning camera, and (c) the rectified version of (b) in terms of each pixel's location.

Instead of computing homogeneous matrices, we can register a pair of images by 3D rotations when we know the internal parameters, such as the actual CCD size and focal length. Two assumptions must be made for PTZ cameras to make this method possible; 1) the optical center is always perpendicular to the CCD sensor, possibly at the center of the image, and this feature is invariant to panning and tilting, 2) the distance between the optical center and the center point is fixed at the same zoom ratio.

The first step for this task is to change the coordinates from the 2D point (X,Y) to a 3D unit vector that has the direction to the 2D point in the image plane from the origin, which is the optical center, with two parameters, θ_p, δ_p, which are shown in Figure 3(a). The center point in the image plane has $\theta_p = 0, \delta_p = 0$, and the focal length and CCD size are necessary to do this transformation. These two angles are

$$\theta_p = \tan^{-1}(X/Z), \quad \delta_p = \tan^{-1}(Y/\sqrt{X^2+Z^2}). \tag{1}$$

Z can be represented by

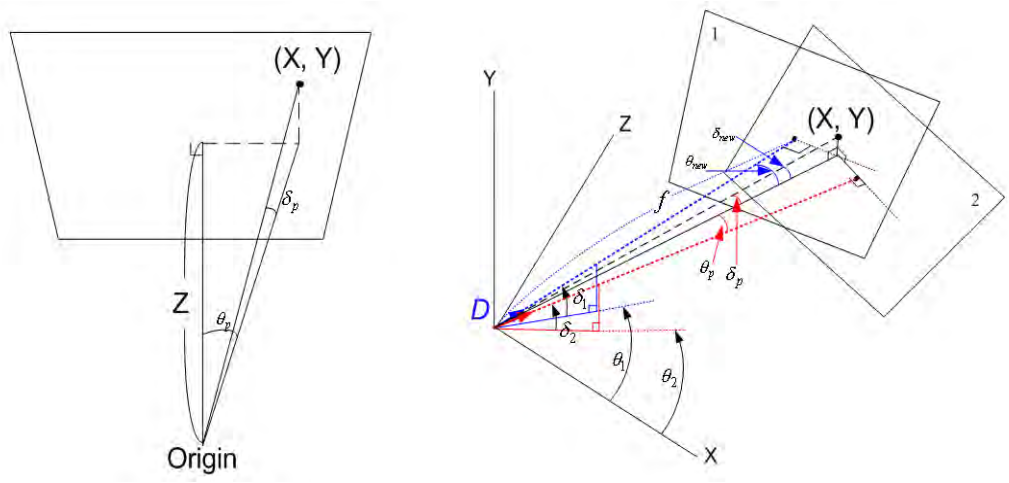
$$Z = \frac{f \cdot X_{width}}{CCD_{width}}, \tag{2}$$

where f is the focal length, CCD_{width} the width of CCD sensor, and X_{width} is the number of total pixels in the horizontal direction in the image plane.

A 3D unit vector can then be defined as

$$\left(\cos\delta_p \sin\theta_p, \sin\delta_p, \cos\delta_p \cos\theta_p\right)^T. \tag{3}$$

This 3D unit vector can represent any point in the image plane, and this vector is invariant to panning and tilting in the camera coordinates, but variant for the world coordinates. In the world coordinates, this vector can be used to represent another image plane that has different pan/tilt angles by applying 3D rotations. Figure 3(b) shows two image planes 1



(a) (b)

Figure 3: Geometric relation between two different camera's views; (a) two angles representing a point in the camera coordinate system, and (b) each image plane can be represented by its pan and tilt angles in the world coordinate system.

and 2 with different pan/tilt angles, such as pan: θ_1 and tilt: δ_1 for image 1, and θ_2 and δ_2 for image 2. (X,Y) in image 2 has a pixel value of the point on an object in the world coordinate in the direction of the 3D unit vector from the origin to (X,Y) in the world coordinate. This point also appears in image plane 1 but the point has a different θ_p,δ_p in the camera coordinate system, compared to image plane 2. By rotating the X and Y axis, we can compute the relation between θ_p,δ_p and $\theta_{new},\delta_{new}$ in the camera coordinates. This can be formulated as

$$\begin{pmatrix} \cos\delta_{New}\sin\theta_{New} \\ \sin\delta_{New} \\ \cos\delta_{New}\cos\theta_{New} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\delta_1 & \sin\delta_1 \\ 0 & -\sin\delta_1 & \cos\delta_1 \end{pmatrix} \begin{pmatrix} \cos(\theta_1-\theta_2) & 0 & \sin(\theta_1-\theta_2) \\ 0 & 1 & 0 \\ -\sin(\theta_1-\theta_2) & 0 & \cos(\theta_1-\theta_2) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\delta_2 & -\sin\delta_2 \\ 0 & \sin\delta_2 & \cos\delta_2 \end{pmatrix} \begin{pmatrix} \cos\delta_p\sin\theta_p \\ \sin\delta_p \\ \cos\delta_p\cos\theta_p \end{pmatrix}. \quad (4)$$

The new vectors, $\theta_{new},\delta_{new}$, can then be solved by

$$\delta_{New} = \sin^{-1}(\sin\delta_{New}),$$
$$\theta_{New} = \sin^{-1}\frac{\cos\delta_{New}\sin\theta_{New}}{\cos\delta_{New}}. \quad (5)$$

So a point (X,Y) in image plane 2 can be mapped into a point (X_{New},Y_{New}) in image plane 1 by the following equations,

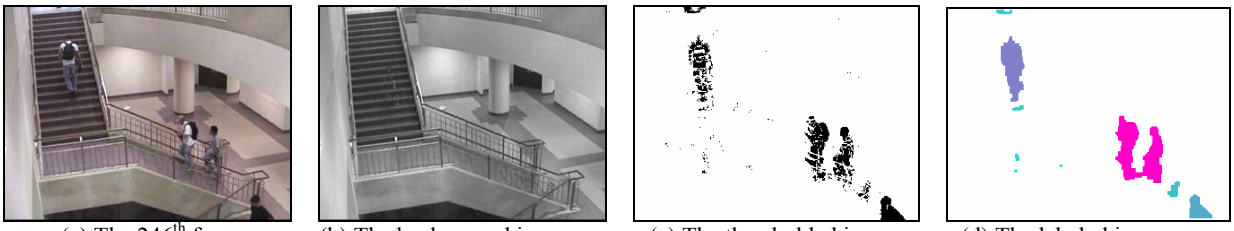
$$X_{New} = Z\cdot\tan\theta_{New},$$
$$Y_{New} = \tan\delta_{New}\cdot\sqrt{X_{New}^2+Z^2}. \quad (6)$$

Every pixel in image 2 will have a corresponding position in image 1, and this means that a generated background image can now be projected into the new view using these relations. For the final system, we defined image plane 1 as the previous view and image plane 2 as the new view. Our algorithm will generate a background for the previous view, and Eq. (6) will generate the corresponding points between the new background and the previous background when the PTZ camera changes angles. We used a nearest neighborhood method to warp the previous background into the new one, so every point's position in the new background with θ_p,δ_p is used to compute the corresponding points in the previous background.

2.3 Moving object detection and Tracking

After a background image is generated, moving pixels can be extracted by comparing the background to the current images. $2T_{BG}$, which is determined experimentally, is used as the threshold value to extract moving pixels, and morphological operations, such as opening and closing, are used to remove noise when detecting moving pixels. Labeling follows the morphological operations.

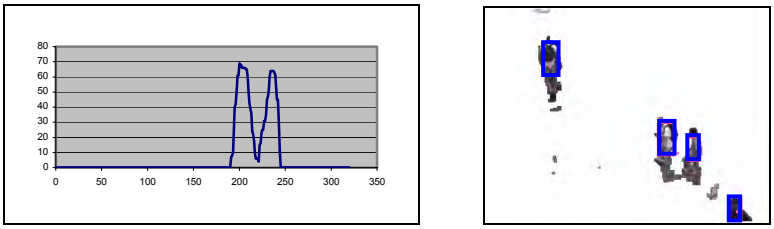
Figure 4(a) shows an input image at the 246th frame and Figure 4(b) the corresponding background image. The thresholded image is shown in Figure 4(c) and the labeled image after morphological operations of Figure 4(c) is shown in Figure 4(d).



(a) The 246th frame (b) The background image (c) The thresholded image (d) The labeled image

Figure 4: An example of background generation, thresholding, and labeling processes.

As shown in Figure 4(d), two objects can be considered as one segment due to noise or shadows. By analyzing the histogram of the vertical projections, the segment can be divided. Figure 5(a) shows the vertical projection histogram, using a threshold value of (0.5*maximum), to determine the number of objects. Figure 5(b) shows the resulting segments that have been separated into two objects, each of which is enclosed by a separate rectangle.



(a) (b)

Figure 5: (a) The vertical projection histogram of the largest segment in Figure 4 (d), and (b) finally detected object areas enclosed by rectangles.

For the tracking, we mainly used a color distribution using histogram intersection[10] and the direction of motion. Movement of humans usually generates deformable motion, which is not easy to track using block-based motion estimation, so features, which are less sensitive to deformation, are necessary. Histogram intersection provides good performance in computing similarities for objects with deformation. Since we use an adaptive background generation and subtraction, we simply compute the histogram of each moving blob and compute the histogram intersection against moving blobs in the previous frame. The result of this operation is the similarity index, which becomes a main criterion for matching. By applying other constraints, such as size and motion that do not change rapidly, we can obtain reasonable tracking performance.

2.4 The implemented System

The implemented system is shown in Figure 6. 320 by 240 images were used for the video input and each image was first used for background generation. If the current pan and tilt angle are unchanged, the output of the projection is the same as the input for this stage obtained by the identity matrices in Equation 4. This background image is then used to extract moving regions by thresholding, followed by morphological operations, which can reduce noise. The next step is labeling and the vertical histogram analysis described in section 2.3 to divide one detected segment that has multiple touching objects into multiple separated parts. The segmented object is finally used to build the color model for tracking, and the final result will have the position of the moving object, color histogram, and center of gravity. The following input image will have the same steps, and the newly detected blob's information will be used for tracking against the previous object information.

The PTZ camera needs to change views for the following reasons; 1) A target can walk or run out of the current view, 2) to facilitate automatic intruder detection for large areas. If the current view changes, the previous and current pan/tilt angles are used to compute 3D rotation matrices as in Equation 4, and this result will be used to generate the Lookup Table (LUT) for rapid projection. There will also be a panning and tilting detection module to determine whether the camera changes its angles for new views. In other words, images captured while the camera changes views will not be

processed because of the uncertainty of the camera's angles, caused by the rapid angle changing mechanism. Once we are certain that the camera is not changing angles and know the current angles, the computed LUT will be used to project the generated background into the new view. Each background will first be enlarged 2X and projected by the nearest neighborhood method, and then resized by 1/2 to the original size to minimize projection errors.

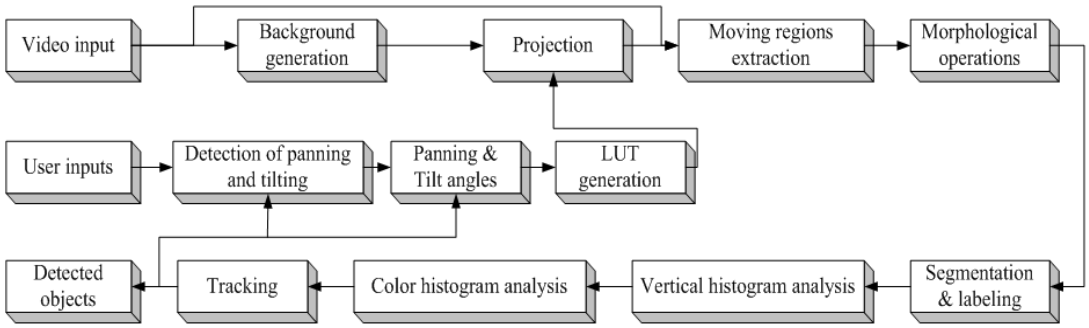


Figure 6: The implemented system.

3. EXPERIMENTAL RESULTS

We mainly tested indoor images for this experiment. A Pentium 4, 1.3GHz PC and a Panasonic PTZ camera were used with a Matrox Meteor II frame grabber. We set the zoom ratio to 1.5X and an initial pan angle: 250˚ and tilt angle: 10˚ were used. The results obtained in testing are shown in Figure 7. We used a timer to change angles by -6˚ for panning, and 1˚ for tilting, and a person walked in each camera's view for testing the algorithms. Figure 7 shows three groups of images; in each group the first row has background images, the second detected moving regions with white, and the third row shows the current image from the PTZ camera with detected moving regions and bounding boxes. We set the background algorithm to update the background image rapidly to get these results, so some of the pixels on a person appear in the generated background. The detected moving regions also contained a white rectangle at right; this region represents the non-overlapping region between the previous view and the current view from panning and tilting.

4. CONCLUSIONS AND FUTURE WORKS

A real-time tracking system using an adaptive background generation and subtraction method was implemented and tested. The system used a mosaicing technique to project one view onto another view with different pan and tilt angles; experimental results were promising for this automatic tracking system using PTZ cameras. Two additional tasks need to be completed to improve performance for detection and tracking; 1) this algorithm is based on intensity values of the background, thus this system can be affected by global intensity variations or the auto-gain-control unit in a PTZ camera, used to maintain overall intensity levels, and 2) when the system changes the camera view, the non-overlapping region, which is the new region created by the camera motion, needs to be updated or substituted with pre-defined images for detecting running targets which can enter this non-overlapping region. A color based algorithm for an adaptive background generation can be used to deal with the first task, or a controlled environment in terms of lighting conditions and camera locations can be the solution. For the second task, we may need a wide angle camera to cover a large area and gain time to update the non-overlapping region before the target enters that area. A wide angle camera, however, can cause lens distortion and the system needs to adapt a routine to remove lens distortion. A strategy on how to change the camera view by certain angles would be another good future task to realize a fully automatic tracking system using multiple PTZ cameras.

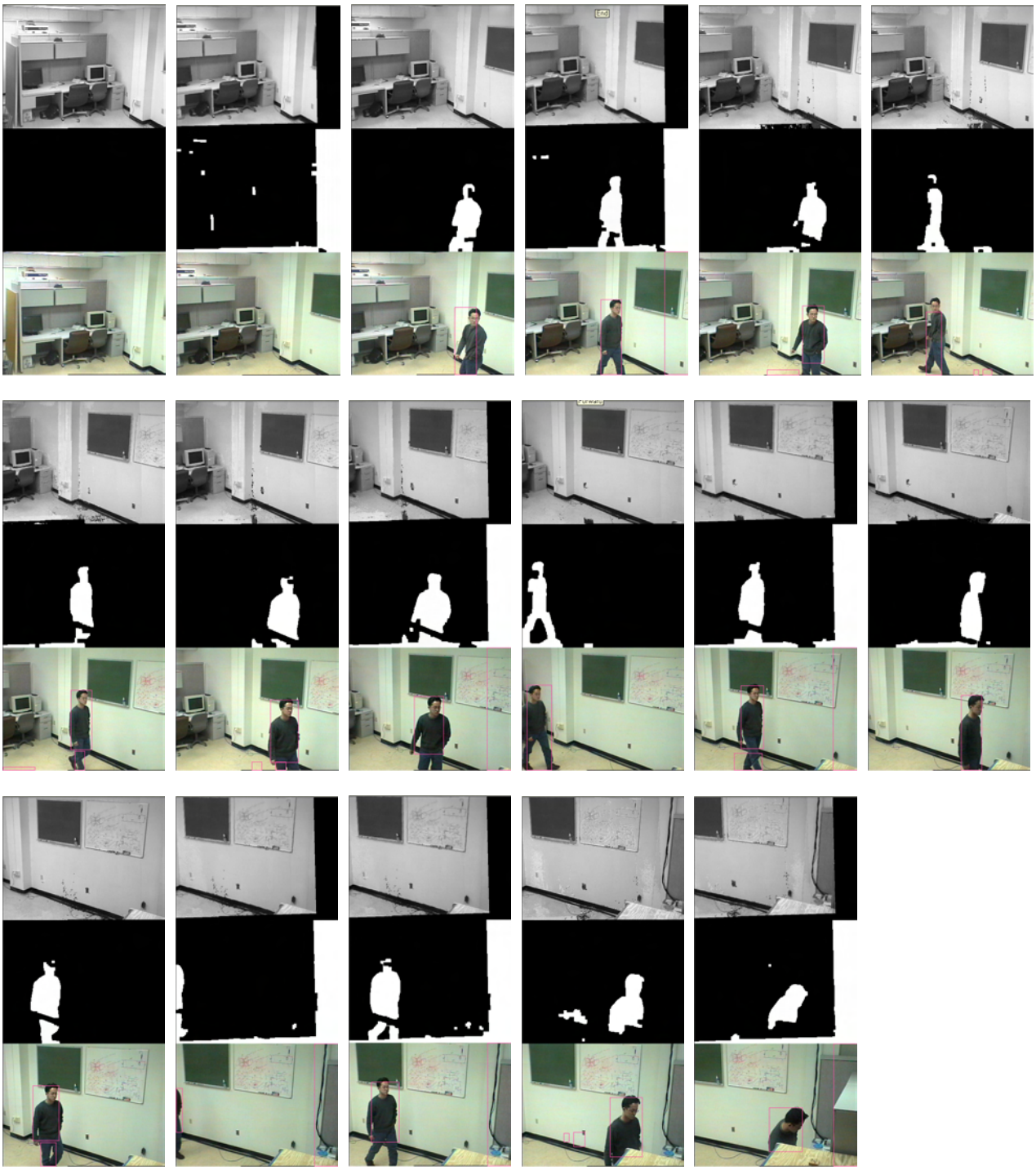


Figure 7: Each group of images has three different types of images; the first row are background images, the second detected moving regions in white, and the third row shows current images from a PTZ camera with detected moving regions.

ACKNOWLEDGEMENTS

This work was supported by TSA/NSSA grant #R011344088.

REFERENCES

1. I. Haritaoglu, D. Harwood and L. S. Davis, “W4: Real-Time Surveillance of People and Their Activities,” IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol.22, No. 8, pp. 809-830, August 2000.
2. F. Dellaert and R. Collins, “Fast Image-Based Tracking by Selective Pixel Integration,” ICCV 99 Workshop on Frame-Rate Vision, September, 1999.
3. A. Blake, M. Isard, *Active Contours*, Springer, London, England, 1998.
4. A. Koschan, S. K. Kang, J. K. Paik, B. R. Abidi, and M. A. Abidi, "Video object tracking based on extended active shape models with color information," Proc. 1st European Conf. Color in Graphics, Imaging, Vision, pp. 126-131, University of Poitiers, France, April, 2002.
5. G. D. Hager and P. N. Belhumeur, “Efficient Region Tracking With Parametric Models of Geometry and Illumination,” IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 20, No. 10, pp. 1025-1039, October 1998.
6. P. Chang and J. Krumm, “Object Recognition with Color Cooccurrence Histograms,” IEEE Conf. on Computer Vision and Pattern Recognition, Fort Collins, CO, June, 1999.
7. T. Horprasert, D. Harwood, and L.S. Davis, “A Robust Background Subtraction and Shadow Detection,” Proc. ACCV'2000, Taipie, Taiwan, January 2000.
8. L. Lee, R. Romano, and G. Stein, “Monitoring activities from multiple video streams: establishing a common coordinate frame,” IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, pp. 758-767, August 2000.
9. M. Nicolescu, G. Medioni, and M. Lee, “Segmentation, tracking and interpretation using panoramic video,” IEEE Workshop on Omnidirectional Vision, pp. 169-174, 2000.
10. M. J. Swain and D. H. Ballard., “Color indexing,” International Journal of Computer Vision, pp. 11-32, 1991.