NOVEL METHODS FOR HIGH-RESOLUTION FACIAL IMAGE CAPTURE USING CALIBRATED PTZ AND STATIC CAMERAS

Richard Yi Da Xu, Junbin Gao, Michael Antolovich

Center for Research in Complex Systems
Charles Sturt University, Australia
{rxu, jbgao, mantolovich}@csu.edu.au

ABSTRACT

In many machine vision applications, a set of static and Pan-Tilt-Zoom (PTZ) cameras are used to capture a sequence of high-resolution facial images of a moving person. In this paper, we present our implementation of such a system. We emphasis two novelties in our work; the first one is our efficient PTZ camera calibration technique using hand-drawn gridlines. The second one is our head position estimation technique using the Gaussian Mixture Model (GMM) and variance analysis of the foreground blob regions.

Index terms: Machine vision, calibration, Image motion analysis

1. INTRODUCTION

The combination of multiple static and PTZ cameras setup is increasingly used in many machine vision applications. A static camera usually has a wide field of view (FOV) and an object-of-interest appears small and within its view most of the times. The PTZ camera on the other hand, has a narrower FOV when zoomed in and the object-of-interest appears to be more detailed.

The examples of such setups can be found in many automated visual surveillance application, where several static cameras are used to observe a scene and one or two PTZ cameras are used for detailed human tracking [1-3]. Another common type of application is the automated instructional video shooting, where a PTZ camera is used to capture the detailed instructor and whiteboard view, while the computer receives and processes the overall lecture information gained from other static cameras [4].

In the case of our high resolution facial image capture system, we essentially need to identify an efficient and yet robust mapping function between:

$$f(\mathrm{x}_0 .. \mathrm{x}_n) \rightarrow (\Lambda_p, \Lambda_t) \qquad (1)$$

where $\{x_i\}$ are a set of image coordinates from the multiple static camera views that represent the centre of the detected face or the head region and Λ_p and Λ_t are the PTZ camera's corresponding pan and tilt values respectively, such that the face would appear in its image center.

The rest of this paper is organized as follows: in section 2, we illustrate our PTZ camera calibration techniques, which allow us to achieve the mapping function in equation 1 robustly. In particular, we emphases our unique method of using Gaussian weighted predication for a robust detection and tracking of grid corners while the PTZ camera undergoes both pan and tilt motions. In section 3, we illustrate our head region detection techniques, where we have applied GMM and variance analysis on the background subtracted regions.

2. EFFICIENT PTZ CAMERA CALIBRATION

A camera's image coordinate x_i and a 3D point X_i are linked by the equation $x_i = K[Rt]X_i$, where the first matrix K contains the camera's intrinsic parameters and second matrix $[Rt]$ contains the extrinsic parameter which is comprised of translation and rotation with respect to a world coordinate frame. The details of a camera's parameters can be found in most computer vision textbooks.

Static camera calibration is achieved relatively easily with the readily available off-the-shelf algorithms [5]. Calibrating a PTZ camera is more challenging. Because each time a camera changes its zoom level, its intrinsic parameters will change and each time its pan and tilt level changes, the extrinsic parameters are affected.

Some modern papers have attempted to use auto (or self) PTZ camera calibration, such as by using the methods described in [3] and [6], where a calibration pattern's geometrical shape is not assumed. However, some auto-calibration methods can lack numerical stability [7], and they rely heavily on the accuracy in detection and tracking of the corresponding points during calibration.

Our system implements the PTZ camera's extrinsic parameter calibration process efficiently and only involves a straightforward setup. We begin by calibrating the intrinsic parameters of all cameras involved using the standard techniques [5].

We then take advantage of the fact that the PTZ camera undertakes pure rotation motions, and the translation of its camera center can be neglected. Equation 1 hence can be divided into three sub-mappings:

978-1-4244-2571-6/08/$25.00 ©2008 IEEE

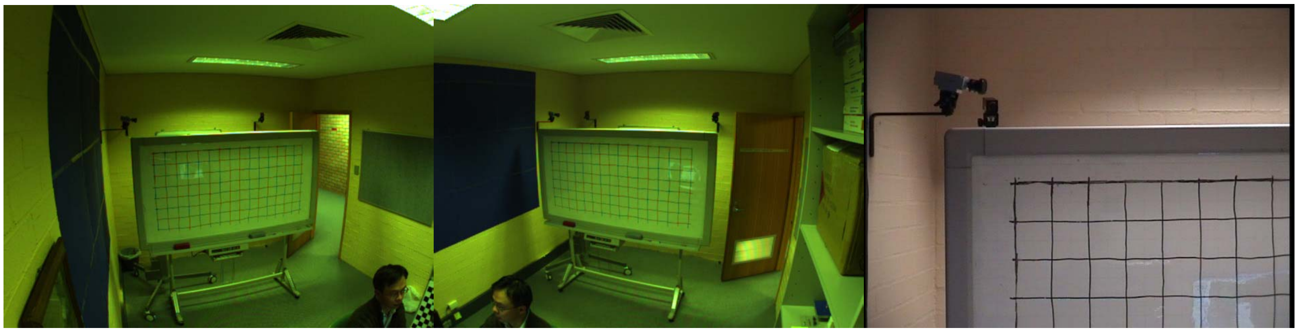

Figure 1: hand-drawn gridlines on a whiteboard. (a) and (b) are from the two static camera views and (c) is the PTZ camera view at its initialized position

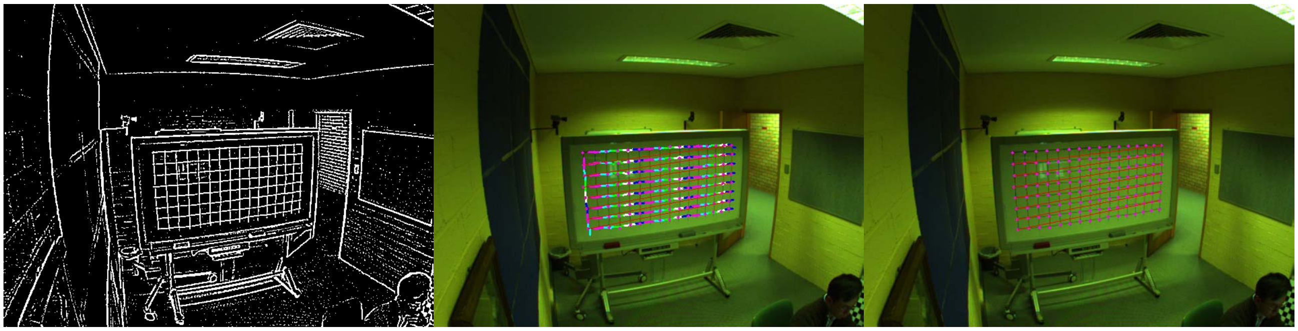

Figure 2: process of grid corner detection (a) the gradient image (b) grid corners and its length segments (c) the detected grid corners

$$f(\mathrm{x}_0 .. \mathrm{x}_n) \rightarrow \mathrm{X_w} \quad (2)$$

$$\mathrm{X_w} \rightarrow (yaw, pitch) \quad (3)$$

$$(yaw, pitch) \rightarrow (\Lambda p, \Lambda t) \quad (4)$$

Equation 2 maps the corresponding image points from the multiple static cameras to a world coordinate $\mathrm{X_w}$. Because the PTZ camera is calibrated to the same coordinate system as the static cameras, $\mathrm{X_w}$ is then used to obtain the corresponding *yaw* and *pitch* angle with respect to the PTZ camera center using a simple geometrical relationship:

$$yaw = \tan^{-1}(X - Cx)/Z - Cz) \quad \text{and}$$

$$pitch = \tan^{-1}(Y - Cy)/Z - Cz) \quad (5)$$

where $(Cx, Cy, Cz)^T$ is the PTZ's camera center.

Finally, equation 4 maps the PTZ camera's rotation angles with its mechanical pan-tilt setting. The rest of this section illustrates our unique implementations to these mappings.

2.1 Calibrate static cameras to a common coordinate

To calibrate the camera's extrinsic parameters, we ask the user to hand draw a set of gridlines on a planar surface, typically on a wall or on a whiteboard, shown in Figure 1. Due to camera noise, lens's radial distortion, room light fluctuation and projective distortion, robust corner detection is not as trivial as it seems. In addition, the PTZ camera view may only contain a partial number of grid corners at one time. To meet these challenges, we have applied several robust detection and tracking techniques. In our work, we ask the user to specify three corners: the leftmost corner and its immediate horizontal and vertical neighboring corners.

The system then detects the rest of the corners using a Gaussian-weighted predication, where we take advantage of the similarities between the two segment (image) lengths on either side of a grid corner.

We begin by obtaining a gradient image using Laplace of Gaussian (LoG) filter (shown in figure 2.a). Using the gradient image, we first perform a line tracking on the left most vertical line and detect all the corners on it. From each of the detected vertical corners, we track its horizontal line and hence to detect the entire set of grid corners.

The line tracking and corner detection is described by equation 6, where the next horizontal pixel location x_n is:

$$x_n = \underset{x \in [x_{n-1} + \frac{1}{2}(x_{n-1} - x_{n-2}) : x_{n-1} + \frac{3}{2}(x_{n-1} - x_{n-2})]}{\arg\max} \int_{x \in W} G(x) \bullet R(x) \quad (6)$$

$$\text{where } G(x) = Ne^{1/2(x-(2x_{n-1}-x_{n-2})/\sigma)^2}$$

In equation 6, x_{n-1} and x_{n-2} are the two previous detected corners and σ is the chosen variance and N is the normalization constant. W is a window size, typically 3 or 5. A similar method is used to detect the next vertical grid corner, where the symbol x_n is replaced by y_n.

R(x) is a 1-d feature, which is the y-component (in the case of vertical grid corner detection, it's the x-component) of the major eigenvector, calculated by modeling all pixel locations (2-d features) with intensity greater than a threshold *t* within a small circular region centered at *(x, y)* using Principal Component Analysis (PCA). Our method is robust against lenses with significant radial distortion. A demo of this work can be viewed at:

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

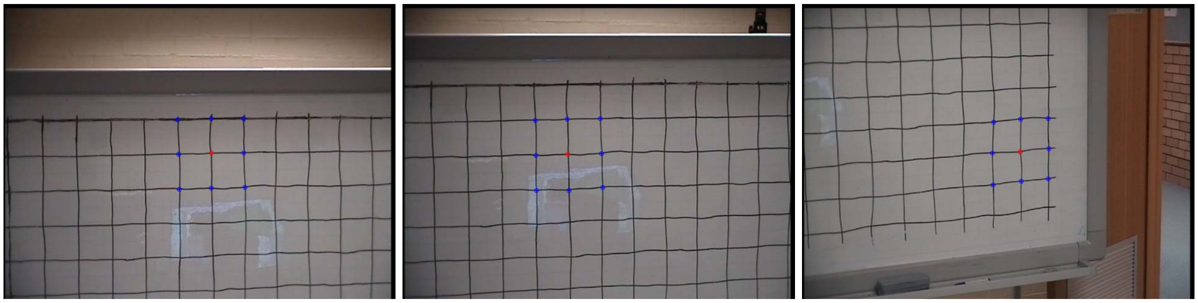

Figure 3: a sequence of images captured from the PTZ camera. The camera undergoes self-adjustment until the 3x3 grid corners are at its image center in each setting. http://silica.csu.edu.au/staff/cs/rxu/videos/ptz_cam.wmv

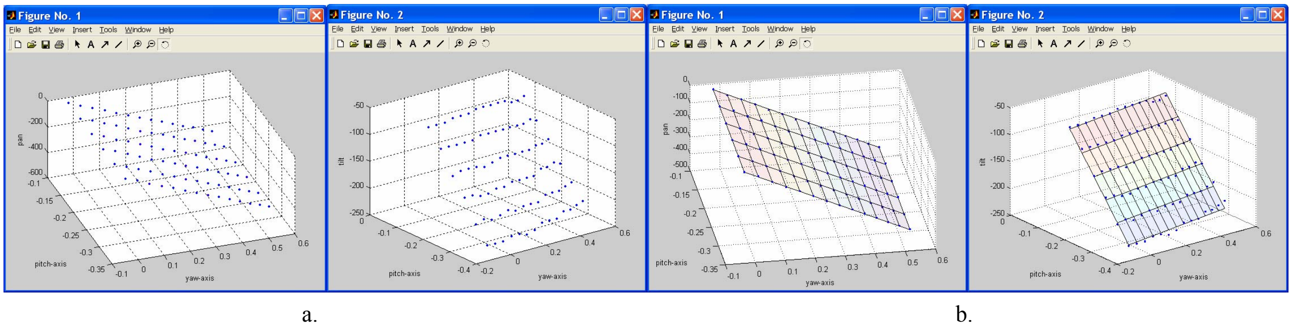

Figure 4: (a) the plotting of ($pitch_i$, yaw_i , pan_i) and ($pitch_i$, yaw_i , $tilt_i$). (b) is the corresponding plane fitting using RANSAC

2.2 Obtaining the PTZ camera mappings

In order to obtain the mapping for equation 3, we must accurately estimate the PTZ camera's "camera center", which requires the knowledge of the PTZ camera's extrinsic parameters at each setting. For equation 4, we need to obtain a sufficient set of corresponding values of *yaw, pitch,* Λ_p and Λ_t in an autonomous fashion.

We use the same algorithm designed for the static cameras (described in section 2.1) to initially detect a 3x3 grid corners. We then program the PTZ camera to repeatedly adjust its pan and tilt values until the midpoint of the tracked grid corners appears at its image center. This process is illustrated in Figure 4 as well as in the accompanying demo video:

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

The grid corner tracking is achieved by forming a set of searching windows, where the estimated corner position of the current video frame is the tracked location of the same corner in the previous video frame. Within each searching window, we compute the corner location using equation 7:

$$q = (\sum \nabla I_{p_i} . \nabla I_{p_i}^{T}).(\sum \nabla I_{p_i} . \nabla I_{p_i}^{T} . p_i) \qquad (7)$$

where ∇I_{p_i} is the image gradient at one of the points p_i in a neighborhood of *q*.

Every time when the midpoint of a 3x3 grid corners appears at the PTZ camera's image center, the entire nine image coordinates and their corresponding 3D positions (we assume 0th z-value for the planar surface) are recorded and used to form a set of extrinsic parameters $[R_i|t_i]$. We estimate the camera center C_i using equation 8:

$$C_i = -(KR_i)^{-1} \bullet (Kt_i) \qquad (8)$$

where K_i is the PTZ camera's intrinsic parameter.

Each C_i differs from each other with a negligible amount at a different pan and tilt setting. In our work, the true value of *C* is chosen to be the median value of all C_i obtained.

We calculate at each setting, a set of yaw_i and $pitch_i$ using equation 5. We then obtain two sets of 3D feature points (yaw_i $pitch_i$, pan_i) and (yaw_i, $pitch_i$, $tilt_i$), shown in figure 4a. We applied a robust plane fitting algorithm using RANSAC to remove the outliers. The result of plane fitting is shown in figure 4.b. With the knowledge of the plane parameters, we can obtain values of Λ_p and Λ_t accurately, given a 3D position X_w.

3. ROBUST FACE REGION DETECTION IN STATIC CAMERA VIEW

Equation 1 shows that in order to compute the 3D coordinate of a person's face, we need to obtain a set of corresponding face points in the multiple static camera views. In many real-time applications, the face/head detection in a large FOV camera view is simplified to be the problem of finding the highest vertical pixel in a foreground blob [4].

Despite its simplicity, this method often achieves better performance than the feature-based face detection approach [8] both in terms of efficiency and robustness. However, the foreground blob method can be problematic when the generated foreground region using background subtraction techniques contains noise particularly associated with shadows. In order to address this issue, we applied a Gaussian Mixture Modal (GMM) with variance analysis:

In our work, we first obtain the foreground using a classical online background training method [9] where each video frames are down-sampled before the statistical modeling.

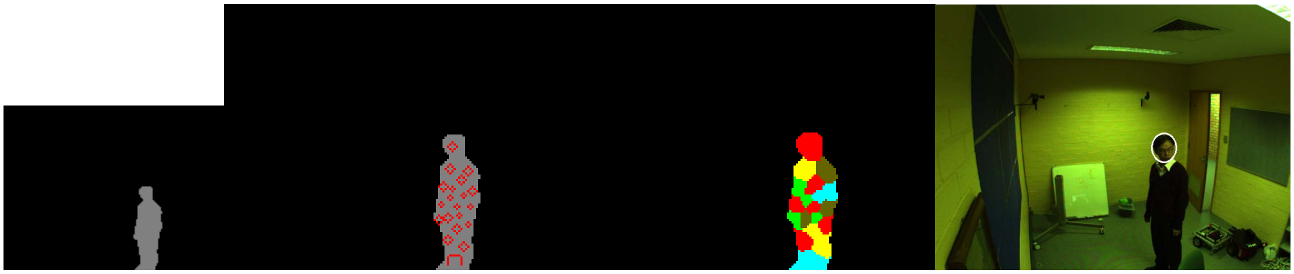
a. b. c. d.

Figure 5: Face/Head region determination process

Next, we use the location data (2d feature) of every pixel belonging to the foreground blob for GMM:

$$\mathrm{gmm}(\mathbf{x}) = \sum_{k=1}^{K} w_k \cdot \frac{1}{\sqrt{2\pi}^{d}\sqrt{\det(\Sigma_k)}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k{}^{-1}(x-\mu_k)} \quad (9)$$

where in order to improve efficiency, we let $\sum = \sigma_k{}^2 I$, and I is a 2D identity matrix, i.e., the variance shape is circular rather than elliptical, as the actual eigenvectors of the covariance matrix $\sum$ are not in our interest.

We let K=20, which is an optimal value determined empirically. At K=20, all of the face region pixels are most likely to belong to the same cluster.

In order for our result to be robust against noise caused by shadows, we reject any cluster with σ_k greater than two times the median σ value of all clusters. We finally claim the cluster mean μ_k with the highest y value in the remaining clusters to be the image points which will be used for equation 2.

Figure 5 illustrate this process: 5.a is the foreground blob. 5.a shows a set of circles with center μ_k and radius σ_k.. 5.c shows foreground pixels belong to each of the clusters. Finally, figure 5.d shows the ellipse fitting using the contours of a cluster with the highest y-value in μ_k.

4. RESULTS AND DISCUSSION

Due to page limit, the screen shots of our facial image capture experiments are not included in the paper. The readers are instead directed to the video demos:

http://silica.csu.edu.au/staff/cs/rxu/videos/demo_work1.wmv

http://silica.csu.edu.au/staff/cs/rxu/videos/demo_work2.wmv

Notice that the face region at most of the times, appears at the center of the PTZ camera view and the detection result is robust under background of complex texture. Our future focus is to combine our existing result with a PTZ camera self-tracking module to leverage the overall tracking performance.

REFERENCES

[1] S.-N. Lim, L. S. Davis, and A. Elgammal, "Scalable image-based multi-camera visual surveillance system," presented at IEEE Conference on Advanced Video and Signal Based Surveillance, 2003, 2003.

[2] A. Hampapur, S. Pankanti, A. Senior, Y.-L. Tian, L. Brown, and R. Bolle, "Face Cataloger: Multi-Scale Imaging for Relating dentity to Location," presented at Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, 2003.

[3] I.-H. Chen and S.-J. Wang, "Efficient Vision-Based Calibration for Visual Surveillance Systems with Multiple PTZ Cameras," presented at IEEE International Conference on Computer Vision Systems, 2006 ICVS '06, 2006.

[4] M. N. Wallick, Y. Rui, and L. W. He, "A portable solution for automatic lecture room camera management," at IEEE Int. Conf. on Multimedia and Exhibition, Taipei, 2004.

[5] R. Hartley and A. Zisserman, *Mutiview Geometry in computer vision*: Cambridge University Press, 2004.

[6] R. Collins and Y. Tsin, "Calibration of an Outdoor Active Camera System," presented at IEEE Computer Vision and Pattern Recognition, 1999.

[7] H. Li and C. Shen, "An LMI Approach for Reliable PTZ Camera Self-Calibration," presented at IEEE International Conference on Video and Signal Based Surveillance, 2006.

[8] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," CVPR 2001, 2001.

[9] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," presented at IEEE CVPR, 1999.