Face Cataloger: Multi-Scale Imaging for Relating Identity to Location

Arun Hampapur, Sharat Pankanti, Andrew Senior,Ying-Li Tian, Lisa Brown, Ruud Bolle
IBM T.J. Watson Research Center
19 Skyline Drive, Hawthorne, NY 10532 USA
{arunh,sharat,aws,yltian,lisabr}@us.ibm.com

ABSTRACT

The level of security at a facility is directly related to how well the facility can keep track of "who is where?" The "who" part of this question is typically addressed through the use of face images for recognition either by a person or a computer face recognition system. The "where" part of this question can be addressed through 3D position tracking. The "who is where" problem is inherently multi-scale, wide angle views are needed for location estimation and high resolution face images for identification. A number of other people tracking challenges like activity understanding are multi-scale in nature.

An effective system to answer "who is where?" must acquire face images without constraining the users and must closely associate the face images with the 3D path of the person. Our solution to this problem uses computer controlled pan-tilt-zoom cameras driven by a 3D wide-baseline stereo tracking system. The pan-tilt-zoom cameras automatically acquire zoomed-in views of a person's head, while the person is in motion within the monitored space.

1. INTRODUCTION

Video surveillance has become increasingly critical to ensuring security. Existing research in this field has taken two distinct directions: the use of biometric identification to answer the 'who' question and video tracking technology to answer the 'where' question. *Our work is focused on building a system that can provide a solution to the combined '**who is where**' question.* The fundamental innovation in this work is the acquisition of face images while the subjects are in motion, at various points along each subject's path through the monitored space. We combine 3D position tracking, with head detection and automatic camera zooming to deliver a catalog of face images, each associated with a unique 3D track. *Clearly, one of the key requirements of such a system is real-time operation, since the zoom cameras need to capture a close-up of the subject.*

Continuous tracking of people provides a significant advantage to identification since we can apply the principle of *continuity of identity* [2]. This says that, while we may only be able to identify a person occasionally (such as when we have a good view of their face, when they swipe an ID badge, or when they speak into a telephone), if we can reliably track the person, we know that all identifications associated with the track relate to the same person and apply throughout the track. Several (fallible) identification methods applied at different times and places can thus be combined and corroborated.

Visual tracking has been a very active area of research[1,4,5,6,8,12,13,15,16]. However there are relatively few efforts which have addressed the issue of multi-scale imaging. Peixoto *et al.* [14] discuss a system, which uses a wide-angle camera to detect people in a scene. It uses a ground plane assumption to infer 3D position of the person. This position is then used to initialize a binocular-active camera to track the person. Optic flow from the binocular camera images is used in smooth pursuit of the target. Stillman *et al.* [17] present a face recognition system for (at most two) people in the scene. They use two static and two pan-tilt-zoom cameras. The static cameras are used to detect people and to estimate their 3D position. This position is used to initialize the pan-tilt-zoom (PTZ) camera. The PTZ -camera images are used to track the target smoothly and recognize faces. The functionality of tracking using the PTZ -camera and face recognition is performed using FaceIt a commercial package from Identix [10].

Collins *et al.* [3] present a wide area surveillance system using multiple cooperative sensors. The goal of the system is to provide seamless coverage of extended areas using a network of sensors. They use background subtraction to detect objects, normalized cross correlation to track targets between frames and classify objects into people and different types of vehicles. Human motion analysis is performed using a star-skeletonization approach. They use both triangulation and the ground plane assumption to determine the 3D position of objects. The camera-derived positions are combined with a digital elevation map. The system has 3D visualization capability for tracked objects and a sophisticated user interface for interacting with the multi-sensor system.

Our approach is to estimate the true 3D position of a person's head using head detection and triangulation, and to use this 3D position to control the pan-tilt-zoom camera. There exists very little work in this area. The approach closest to our work is by Stillmann *et al.* [17].

Our approach uses shape-based head detection to establish correspondence between the two static views. Stillman *et al.* use a narrow stereo baseline and color correlation to establish correspondence. They also use face tracking in the fine view to keep the face centered in the high-resolution view. Our approach of using the coarse scale tracking to drive the PTZ cameras is more robust since losing track at the higher resolution is more likely. Their approach does not scale to multiple people and other body parts, both of which we are poised to address.

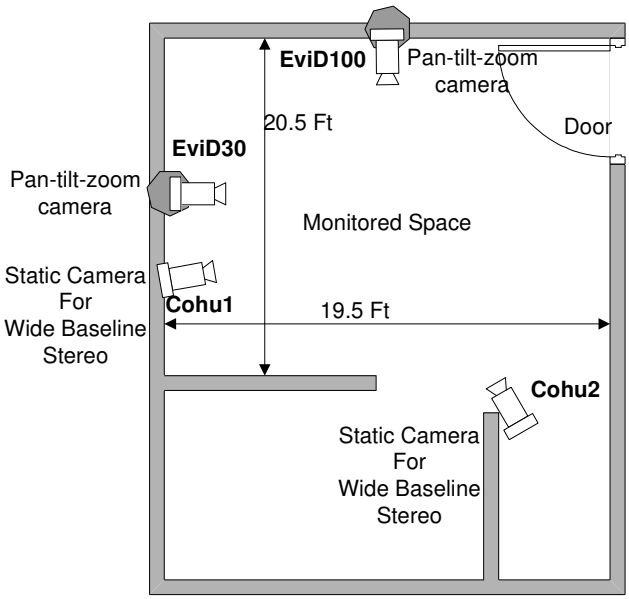

Figure 1 Plan view of the monitored space, with two fixed cameras (Cohu1, Cohu2) and two pan-tilt-zoom cameras (EviD30, EviD100).

2. SYSTEM OVERVIEW

Figure 1 shows the camera setup for the face cataloging system. The two static cameras (Cohu 1000) have overlapping fields of view and are used for wide baseline stereo triangulation. The two pan-tilt-zoom cameras (Sony EviD30, EviD100) are used to zoom in on the moving targets. All the cameras, both static and pan-tilt are *calibrated to a common coordinate* system. We have used the OpenCV [11] camera calibration code. The monitored space is about 20ft x 19ft. The tracking and camera control components of the face cataloging system run real time on a dual 2GHz Pentium machine. The video recorder is on a separate server which communicates with the tracking server via a socket interface. Sample videos of the system can be found at http://www.research.ibm.com/people/a/aws/peoplevision/videos.html

Figure 2 shows a high level block diagram of the face cataloging system. Each of the static camera images is segmented using background subtraction [9]. The foreground blobs are tracked by the 2D Multi-blob Tracker. The outputs of the two 2D trackers are combined by the 3D Multi-Blob Tracker to generate 3D tracks. Each 3D track is analyzed to detect the head in the component 2D views. The 2D head positions are used to determine the true 3D position of the persons head. The 3D head positions are then used by the active camera manager to assign the cameras to appropriate tracks and drive pan-tilt-zoom parameters of the cameras. There are a variety of policies for active camera assignment and control. The details of each of the major components are described in the following sections.

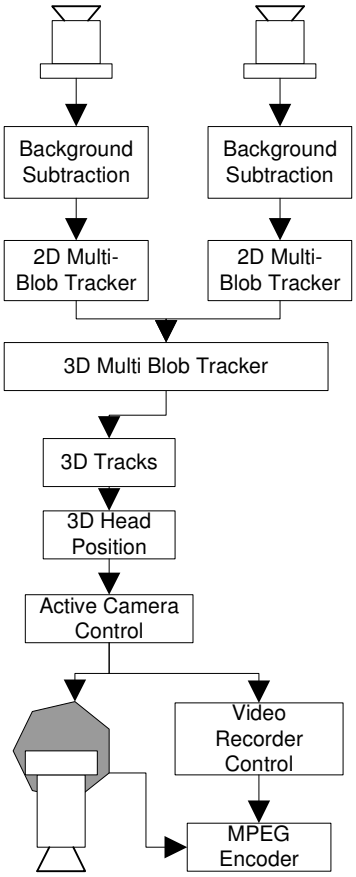

Figure 2: Block diagram of face cataloger system.

3. 3D MULTI-BLOB TRACKING

Figure 3 shows the key components of the 3D multi-blob tracking system. The 2D blob tracking relies on appearance models which are image templates. New appearance models are created when an object enters a scene. In every new frame, each of the existing tracks is used to try to explain the foreground pixels. The fitting mechanism used is correlation, implemented as minimization of sum of absolute pixel differences over a predefined search area. During occlusions, foreground pixels may be overlapped by several appearance models. Color similarity is used to determine which model lies in front and infer a relative depth ordering for the tracks.

Once this relative depth ordering is established, the tracks are correlated in order of depth. The correlation process is gated by the explanation map which holds at each pixel the identities of the tracks explaining the pixels. Thus foreground pixels that have already been explained by a track do not participate in the correlation process with more distant models. The explanation map is now used to update the appearance models of each of the existing tracks. Regions of foreground pixels that are not explained by existing tracks are candidates for new tracks. A detailed discussion of the 2D multi-blob tracking algorithm can be found in [18]. *The 2D multi-blob tracker is capable of tracking multiple objects moving within the field of view of the camera, while maintaining an accurate model of the shape and color of the object.*

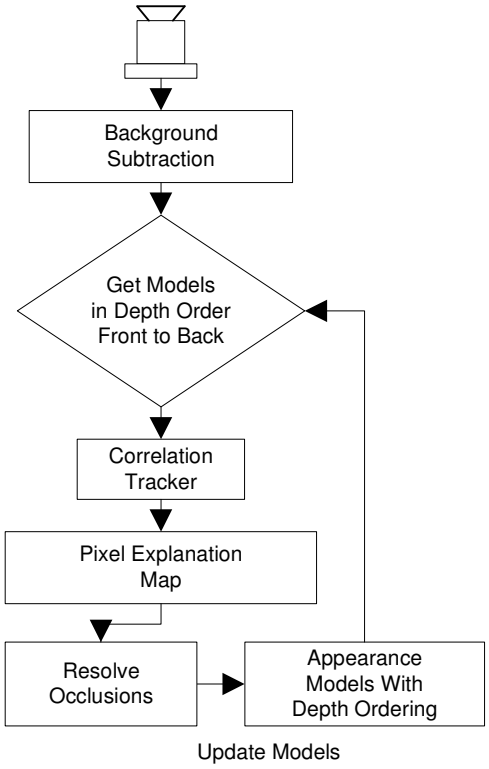

Figure 3 Block Diagram of 2D Multi-Blob Tracker.

Figure 4 shows a block diagram of the 3D tracker which uses wide baseline stereo to derive the 3D positions of objects. At every frame, we measure the color distance between all possible pairings of tracks from the 2 views. We use the Bhattacharya distance between the normalized color histograms of the tracks. For each pair we also measure the triangulation error, which is defined as the shortest 3D distance between the rays passing through the centroids of the appearance models in the two views. The triangulation error is generated using the camera calibration data. To establish correspondence we minimize the color distance between the tracks from the view with the smaller number of tracks to the view with the larger number. This process can potentially lead to multiple tracks from one view being assigned to the same track in the other. We use the triangulation error to eliminate such multiple assignments. The triangulation error for the final correspondence is thresholded to eliminate spurious matches that can occur when objects are just visible in one of the two views. Once a correspondence is available at a given frame, we now need to establish a match between the existing set of 3D tracks and 3D objects present in the current frame. We use the component 2D track identifiers of a 3D track and match them against the component 2D track identifiers of the current set of objects to establish the correspondence. The system also allows for partial matches, thus ensuring a continuous 3D track even when one of the 2D tracks fails. *Thus the 3D tracker is capable of generating 3D position tracks of the centroid of each moving object in the scene. It also has access to the 2D shape and color models from the two views that make up the track.*

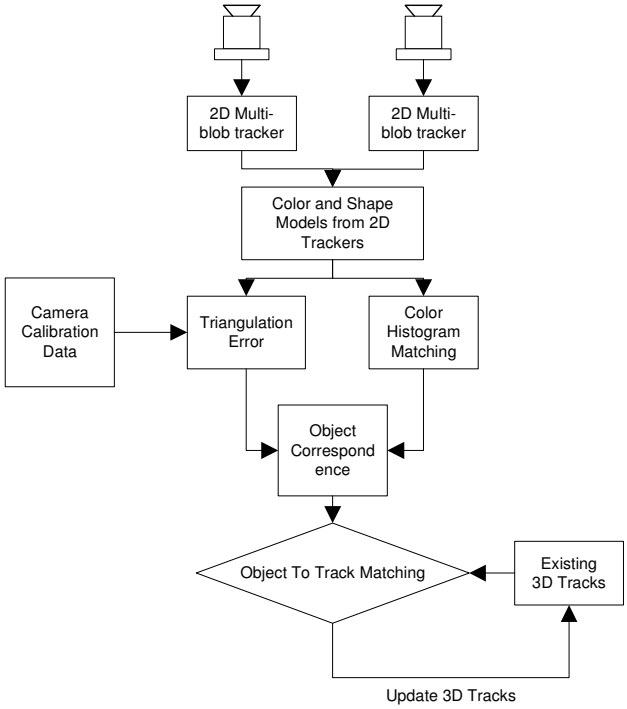

Figure 4 Block Diagram of 3D Multi-blob tracker.

4. HEAD DETECTION

The head detection uses the smoothed silhouette of the foreground object as segmented using background subtraction. To interpret the silhouette, we use a simple human body model consisting of six body parts: head, abdomen, two hands, and two feet. First, we generate a one-dimensional "distance profile" that is the distance of each contour pixel from the contour centroid, following the contour clockwise. This distance profile is parsed into peaks and valleys based on the relative magnitudes of the successive extrema. The peaks of the distance transform are used to hypothesize candidate locations of the five

body parts: the head, two feet, and two hands. Determination of the head among the candidate locations is currently based a number of heuristics based on the relative positions of the candidate locations and the curvatures of the contour at the candidate locations. More specifically, the following objective function is used to decide the location of the head:

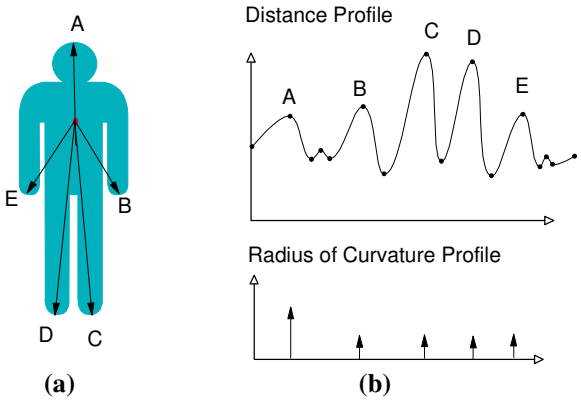

Figure 5. Head detection steps. (a) The silhouette information (b) Distance profile showing significant peaks and the radii of curvature at the significant peaks.

$$O_i = (Y_i - Y_c) + w_x * |(X_c - X_i)| + w_r * R_i - w_e * E_i,$$

where (X_c, Y_c), (X_i, Y_i) denote the co-ordinates of the centroid of the body contour and center of the circle fitted to the contour segment associated with i^{th} peak. R_i, E_i denote radius and residue of least square fitting of the i^{th} circle. $w_x(=1), w_r(=1)$, and $w_e(=10)$ are weights associated with three components of the objective function. In other words, the objective function hypothesizes that smaller, more circular extrema are more likely to be heads. Similarly, the circles that are higher and vertically more aligned with the center of the body are preferred as heads. Our approach is similar to [7].

5. ACTIVE CAMERA MANAGEMENT

There are two components in the active camera manager: the camera parameter controller (CPC) and the camera assignment manager (CAM). The CAM is charged with the responsibility of assigning the fixed number of pan-tilt-zoom cameras to the objects that are active within the monitored space. Given that an active camera has been assigned to acquire close-up views of an object, the CPC is in charge of controlling the pan-tilt-zoom parameters of the camera on an ongoing basis.

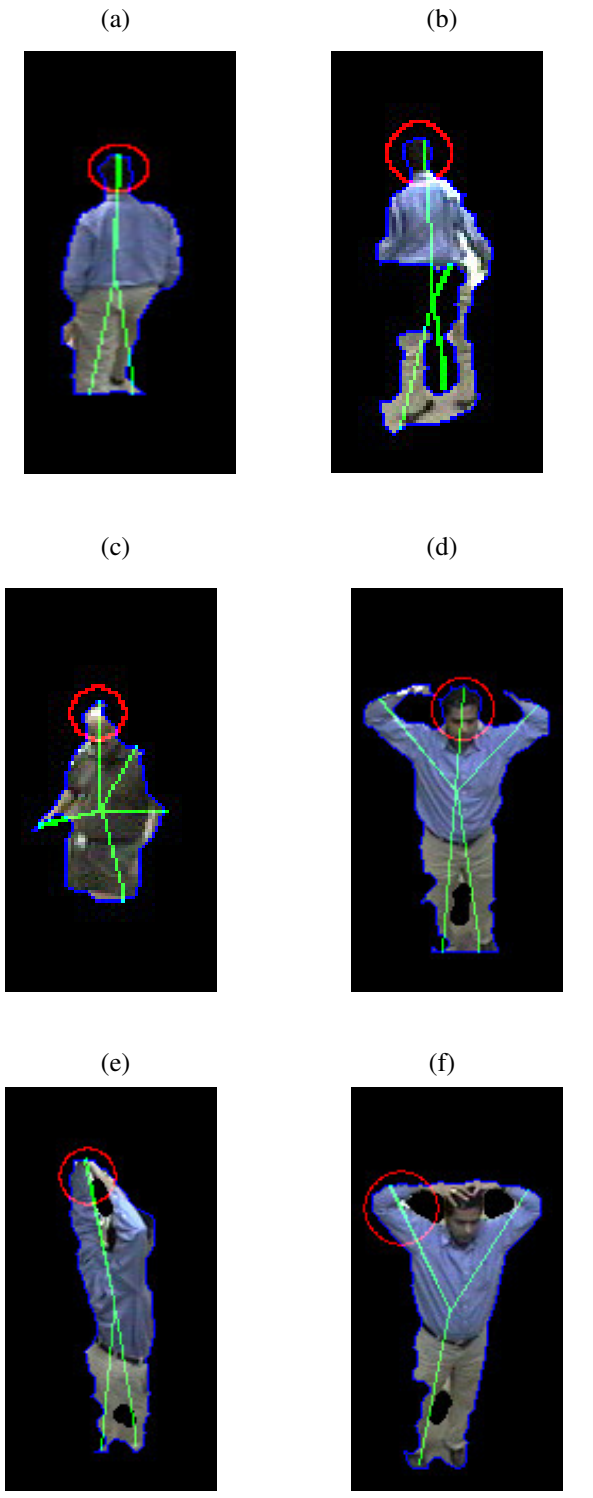

Figure 6 Output of the head detector on sample foregrounds obtained from the background subtraction. (a)-(d) show successful detection where as (e), (f) illustrate failure modes.

Camera Assignment Manager:
The camera assignment manager is essentially a resource allocation algorithm. The resource allocation problem is made simpler when the number of active cameras is greater than the number of currently active tracks, but in all cases a number of different policies can be followed for assigning cameras to the subjects in the monitored space. The choice of policy will be driven by the application goals, following are few examples.

- **Location-Specific Assignment:** Here the active cameras are assigned to objects moving near certain locations within the monitored space. For example, zoom in on persons near entrances.
- **Orientation-Specific Assignment:** Here people are assigned cameras in front of them so that the clearest view of each person's face is obtained.
- **Round Robin Sampling:** Here the cameras are periodically assigned to different objects within the scene with the goal of uniformly covering all objects with close-up views.
- **Activity Based Assignment**: Here the cameras can be assigned to people or objects performing certain activity. For example, in an airport, active cameras could automatically be assigned to track people who are running.

Camera Parameter Control:
This module is in charge of controlling the pan, tilt and zoom parameters of the camera. The objective of the camera control is to maintain the person being tracked within center of its view and to provide a closer view of the person. We use 3D position and velocity of the person for steering the pan tilt zoom camera. The pan and tilt of the camera are controlled to position the detected head location at the center of the active camera image. Ideally, one would like to zoom the camera to maximize of portion of the image depicting head. This is not feasible because it entails tracking speeds not achievable by a typical low-cost steerable camera, especially, when the person is moving briskly close to the camera. The exact relationship between the effective zoom value, z, and the tracked person is governed by

$$z = z_{\min} + (z_{\max} - z_{\min}) * s(v, v_t) * f\left(\frac{d - d_{\min}}{d_{\max} - d_{\min}}\right),$$

where $[d_{\min}, d_{\max}]$, $[z_{\min}, z_{\max}]$ denote the ranges of distances in the operating space and camera zooms. d is distance of the person from the camera and v is speed of the person. $f(.)$, $s(.,.)$ represent distance and speed based zoom modulating functions. In our system, $f(.)$ is implemented as a continuous function and can be one of the following two policies: (a) linear, when $f(x) = x$. Or (b) sublinear, when $f(x) = \sqrt{x}$. On the other hand, we have implemented $s(v, v_t)$ as a discrete function:

$$s(v, v_t) = \begin{cases} m_c, v < v_t \\ m_w, v \geq v_t \end{cases}$$

where, v_t is a speed threshold, and m_c, m_w denote zoom multipliers corresponding to close-up and wide-angle views of the scene. That is, the system steers the zoom of the camera in two modes: (a) *Close-up*: when the person is deemed to be static or moving with sufficiently small velocity, the zoom multiplier of the camera is set to a predetermined high value $(m_c = 0.1)$; (b) *Wide Angle*: when the person is moving with higher speeds, the zoom multiplier is set to predetermined low value $(m_w = 0.0)$. Figure 7 shows the zoom values switching to close-up at low velocity points.

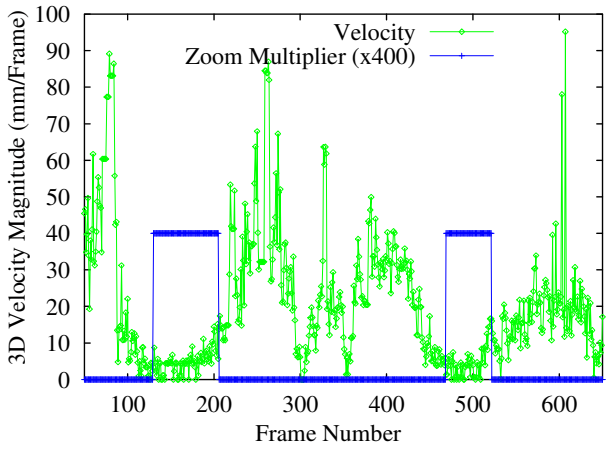

Figure 7: Zoom control signal and relationship to measured head velocity.

6. ERROR ANALYSIS

The ultimate goal of the face cataloger is to obtain good close-up head shots of people walking through the monitored space. The quality of the close-up face clips is a function of the accuracy of a number of underlying components. Following are the potential sources of errors in the system.

- **Tracking Continuity Errors**: These are errors in the continuous tracking of objects, these could be
  - 2D Track Breakage: These errors occur when the tracker prematurely terminates a track and creates a new track for the same object.
  - 2D Track Swap: This error occurs when the objects being represented by a track

get interchanged, typically after an occlusion.

- 3D Track Swap: This can occur due to errors in the inter-view correspondence process.

- **2D Head Detection Errors**: These are errors in the position and size of the head detected in each of the 2D views.
- **True Head Center Error**. Since we are detecting the head in two widely different views, the centers of the two head bounding boxes do not correspond to a single physical point and hence will lead to errors in the 3D position.
- **3D Head Position Errors**: There are errors in the 3D position of the head due to inaccuracy in the camera calibration data.
- **Active Camera Control Errors:** These are errors that arise due to the active camera control policies. For example, the zoom factor of the camera is dependent on the velocity of the person, thus any error in velocity estimation will lead to errors in the zoom control.
- **Active Camera Delays**: The delay in the control and physical motion of the camera will cause the close-up view of the head to be incorrect.

7. EXPERIMENTS

The errors discussed in the previous section break down into two distinct classes, errors in multi object tracking, and errors in acquiring close-up face images. In this paper we focus on the evaluation of the ability to acquire close-up images of the face. Early performance metrics of our 2D multi-object tracking system have been reported in [18]. Sample results on the multi-object tracking can be seen on our web page http://www.research.ibm.com/people/a/aws/peoplevision/videos.html.

Head Close-Up Performance: Figure 8 shows the results of a sample run, where a person walked through the space. Figure 9 shows the corresponding static and close-up camera images at two positions along the path. This video was acquired using the sub-linear zoom policy discussed above. Clearly *the close-up images have much more information relating to identity*. These images however are not yet suitable for current day face recognition algorithms which require a fairly frontal view of the face. The acquisition of frontal facial images requires face orientation estimation and appropriate deployment of pan-tilt-zoom cameras. This is one of the enhancements that are planned to the system presented herein.

.

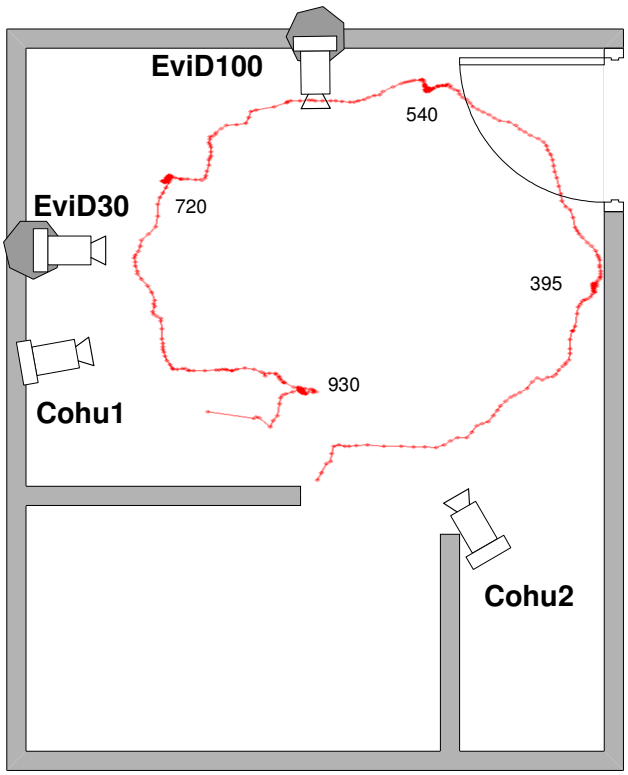

Figure 8: Plot of the X,Y positions of the persons path through the monitored space. Number along the path indicate positions where the "velocity control policy" zoomed in on the person. Images at two of these points are shown in Figure 9.

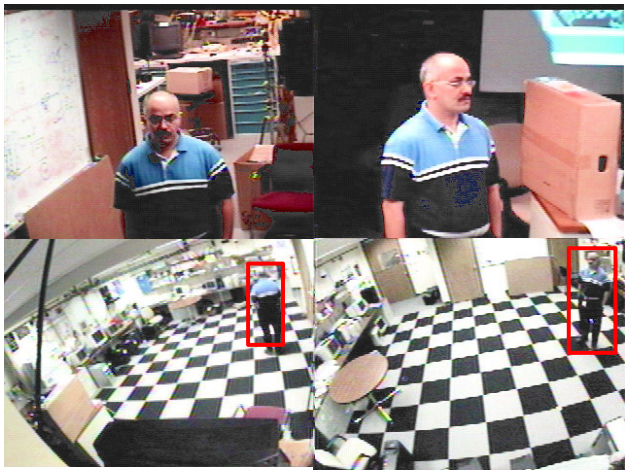

Figure 9: <u>Left Top:</u> Image from EviD100 at frame 395. <u>Left Bottom:</u> Image from Cohu2 at frame 395. <u>Right Top:</u> Image from EviD100 at frame 930. <u>Right Bottom:</u> Image from Cohu1 at frame 930.

Experimental Procedure: The basic experiment for all the results on head close-up performance involves a single person walking around the monitored space. In all these experiments the zoom was set a fixed value of

approximately 1/3rd of the maximum allowable zoom for both the PTZ cameras. The following are the steps.

1. Run the face cataloger with one person walking around the monitored space.
2. Both the outputs of the PTZ cameras and the static cameras are saved as AVI files, which are then used for ground truth marking.
3. Manually annotate the position of the head in the close-up views and the zoomed in views at regular intervals through the sequence. The head is marked using a bounding rectangle on a GUI.

2D Head Detection Performance

Here we measure the distance between the centroids of the bounding boxes of the head in the static view as marked by the human annotator and as detected by the head detection algorithm. Figure 10 shows the errors in pixel position over a single run and Figure 11 shows the error distribution over multiple runs. Clearly, the head detection process is very accurate for normal activities like walking. Dealing with adversarial behavior will require further work.

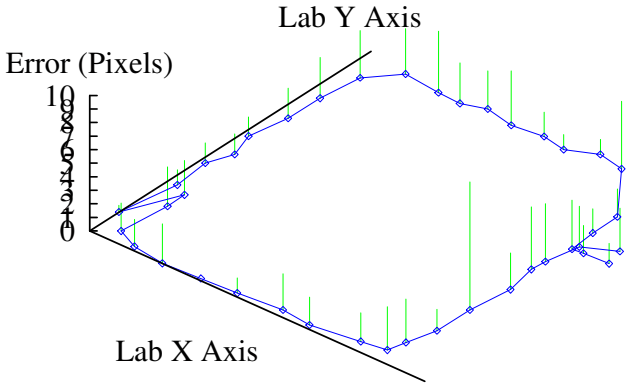

Figure 10 Pixel distance between the head centroid position reported by the head detector and annotated ground truth for a sample run.

3D Head Position Detection Performance:

The primary source of this error is camera calibration. A good measure of this error is the distance between the center of the head (as detected by the 2D head detection system) and the re-projected position of the head from 3D, after triangulation. If the calibration data were perfect, the re-projected point would coincide with the 2D head centroid. Figure 12 shows the pixel distance between the 2D head position and the re-projected position at each point along the persons trajectory. Clearly the error is not uniform at all parts of the space and is a function of the relative position with reference to the two static cameras. Figure 13 shows the re-projection error distribution. Better calibration can improve the performance of the system.

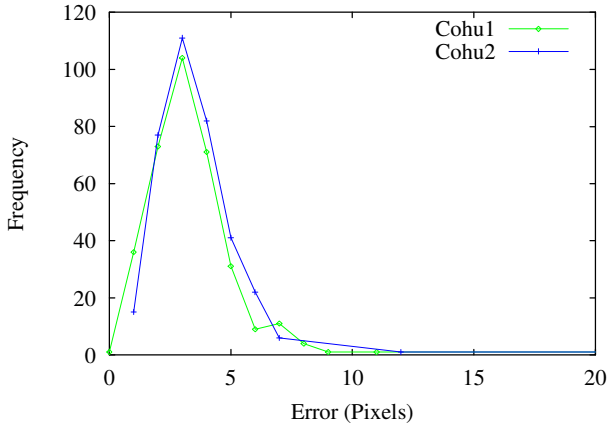

Figure 11 Head position error (from ground truth) distribution over multiple runs. Errors reported for both static cameras.

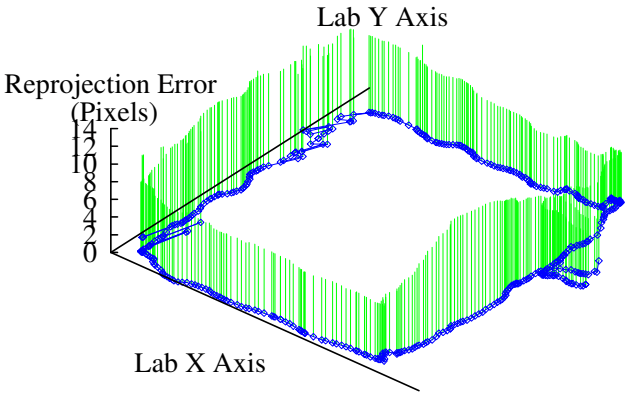

Figure 12: Pixel distance between the head centroid position reported by the head detector and the re-projection of the 3D head position onto one of the static views.

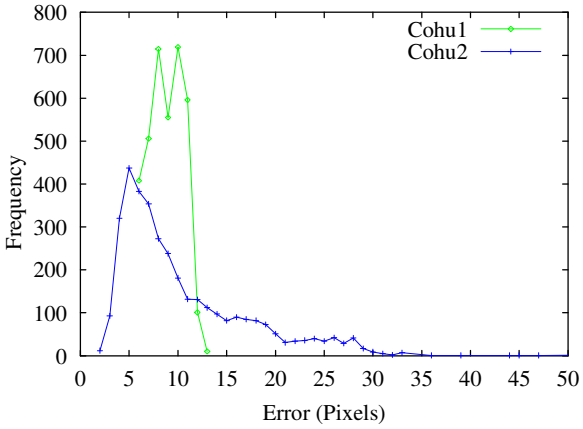

Figure 13 3D Head re-projection error distribution over multiple runs. Errors reported for both static cameras.

Close-up Head Capture Performance:

Figure 14 shows how the probability of head capture as a function of the zoom factor. This was computed by measuring the distance of the head (as marked by the annotator) in the close-up views from the centre of the close-up image at the default zoom value. These measurements are then used to generate the probability at higher zoom factors by appropriately scaling the size of the image and determining if the entire bounding box of the head is contained within the zoomed in view. The EviD100 has faster mechanical motion than the EviD30, this is clearly visible in the plot.

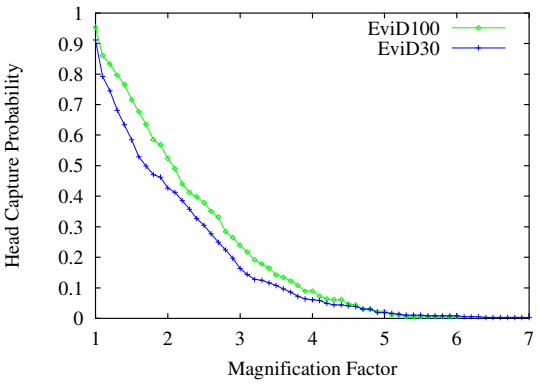

Figure 14: Probability of head capture as a function of zoom

8. CONCLUSIONS

This paper has presented a novel system for linking identity to location for security and surveillance applications. The system uses active cameras to zoom in on the head region of people as they move freely about the monitored space. The paper presented the first set of evaluations of the face cataloging system. Our plans include long term evaluations of the system, evaluations of camera control policies and evaluation in the targeted environments (like airports, stores etc).

Clearly, achieving high levels of security at a facility is a complex challenge of which technology is one of the components. The system presented herein beings to address one of the critical challenges of video surveillance, namely the ability to selectively focus attention of the system and to acquire information at multiple scales. Face cataloging is one instantiation of multi-scale imaging. A similar system could be used to acquire close-up videos of suspicious activities, or zoomed in pictures of cars on a freeway.

REFERENCESb

1. Anjum Ali, J. K. Aggarwal: Segmentation and Recognition of Continuous Human Activity. IEEE Workshop on Detection and Recognition of Events in Video 2001.
2. R. M. Bolle, J. H. Connell, S. Pankanti, N. K.Ratha, A.Senior, Biometrics 101, IBM Research Report, Computer Science, RC22481, June 2002.
3. Collins, Lipton, Fujiyoshi, and Kanade, "Algorithms for cooperative multisensor surveillance," Proc. IEEE , Vol. 89, No. 10, Oct. 2001.
4. D. Comaniciu, V. Ramesh, P. Meer: *Real-Time Tracking of Non-Rigid Objects using Mean Shift*,, IEEE Conf. Computer Vision and Pattern Recognition (CVPR'00), Hilton Head Island, South Carolina, Vol. 2, 142-149, 2000
5. Trevor Darrell, David Demirdjian, Neal Checka, Pedro Felzenszwalb: Plan-View Trajectory Estimation with Dense Stereo Background Models. ICCV 2001: 628-635.
6. S. Dockstader and A. M. Tekalp, ``Feature extraction for the analysis of gait and human motion," Proc. ICPR, Quebec City, Quebec, August 2002.
7. H. Fujiyoshi and A. Lipton, Real-time Human Motion Analysis by Image Skeletonization, Proc. of the Workshop on Application of Computer Vision, October, 1998.
8. Haritaoğlu and Flickner, "Detection and Tracking of Shopping Groups in Stores," CVPR 2001.
9. T. Horprasert, D. Harwood, and L. Davis. A Statistical Approach for Real-Time Robust Background Subtraction and Shadow Detection. Proceedings of IEEE Frame-Rate Workshop, Kerkyra, Greece, 1999.
10. Identix. http://www.identix.com/ Formerly Visionics Corporation
11. Intel. Open Source Computer Vision Library (OpenCV), http://www.intel.com/research/mrl/research/opencv.
12. I. Pavlidis and T. Faltesek, A Video-Based Surveillance Solution for Protecting the Air-Intakes of Buildings fromChem-Bio Attacks, ICIP 02.
13. G. Kogut, M. Trivedi, "A Wide Area Tracking System for Vision Sensor Networks," 9th World Congress on Intelligent Transport Systems, Chicalgo, Illinois,October,2002.
14. Peixoto, Batista and Araujo, "A Surveillance System Combining Peripheral and Foveated Motion Tracking," ICPR, 1998.
15. G. Qian, R. Chellappa, and Q. Zheng, A Bayesian Approach to Simultaneous Motion Estimation of Multiple Independently Moving Objects, International Conference on Pattern Recognition, Quebec City, Canada, I.9, 2002.
16. Paolo Remagnino, Graeme A. Jones , Nikos Paragios ,Carlo S. Regazzoni Video-Based Surveillance Systems: Computer Vision and Distributed Processing.
17. Stillman, Tanawongsuwan and Essa, "A System for Tracking and Recognizing Multiple People with Multiple Cameras," Georgia TR# GIT-GVU-98-25, August 1998.
18. A. Senior, A. Hampapur, Y-L. Tian, L. Brown, S. Pankanti, and R. Bolle. Appearance Models for Occlusion Handling. Proceedings of Workshop on Performance Evaluation of Tracking and Surveillance (PETS2001), December 2001.

.