Tracking Multiple Occluding People by Localizing on Multiple Scene Planes

Saad M. Khan and Mubarak Shah, *Fellow*, *IEEE*

Abstract—Occlusion and lack of visibility in crowded and cluttered scenes make it difficult to track individual people correctly and consistently, particularly in a single view. We present a multiview approach to solve this problem. In our approach, we neither detect nor track objects from any single camera or camera pair; rather, evidence is gathered from all of the cameras into a synergistic framework and detection and tracking results are propagated back to each view. Unlike other multiview approaches that require fully calibrated views, our approach is purely image-based and uses only 2D constructs. To this end, we develop a planar *homographic occupancy constraint* that fuses foreground likelihood information from multiple views to resolve occlusions and localize people on a reference scene plane. For greater robustness, this process is extended to multiple planes parallel to the reference plane in the framework of plane to plane homologies. Our fusion methodology also models scene clutter using the Schmieder and Weathersby clutter measure, which acts as a confidence prior, to assign higher fusion weight to views with lesser clutter. Detection and tracking are performed simultaneously by graph cuts segmentation of tracks in the space-time occupancy likelihood data. Experimental results with detailed qualitative and quantitative analysis are demonstrated in challenging multiview crowded scenes.

Index Terms—Tracking, sensor fusion, graph-theoretic methods.

1 Introduction

Tracking multiple people accurately in cluttered and crowded scenes is a challenging task primarily due to occlusion between people. If a person is visually isolated (i.e., neither occluded nor occluding another person in the scene), it is much simpler to perform the tasks of detection and tracking. This is because the physical attributes of the person's foreground blob, like color distribution, shape, and orientation, remain largely unchanged as he/she moves. Increasing the density of objects in the scene increases interobject occlusions. A foreground blob is no longer guaranteed to belong to a single person and may belong to several people in the scene. Even worse, a person might be completely occluded by other people. Under such conditions of limited visibility and clutter, it might be impossible to detect and track multiple people using only a single view. The next logical step is to use multiple views of the same scene in an effort to recover information that might be missing in a particular view. In this paper, we propose a multiview approach to detect and track multiple people in crowded and cluttered scenes. We are interested in situations where the scene is sufficiently dense that partial or total occlusions are common and it cannot be guaranteed that any person will be visually isolated. Fig. 1 shows several examples of crowded scenes that we used to test our approach. Notice that very few people are viewed in isolation and there are cases of near total occlusion.

• *The authors are with the Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816. E-mail: skhan@sarnoff.com, shah@eecs.ucf.edu.*

Manuscript received 17 July 2007; revised 19 Feb. 2008; accepted 24 Mar. 2008; published online 15 Apr. 2008.
Recommended for acceptance by J. Luo.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-07-0434.
Digital Object Identifier no. 10.1109/TPAMI.2008.102.

In our approach, we do not use color models or shape cues of individual people. We neither detect nor track objects in any single camera, or camera pair; rather, evidence is gathered from all the cameras into a synergistic framework, and detection and tracking results are propagated back to each view. Our method of detection and occlusion resolution is based on geometrical constructs and requires only the distinction of foreground from background, which is obtained using standard background modeling techniques. At the core of our method is a planar homographic occupancy constraint [27] that combines foreground likelihood information (probability of a pixel in the image belonging to the foreground) from different views to resolve occlusions and determine regions on scene planes that are occupied by people. The homographic occupancy constraint interprets foreground as scene occupancy by nonbackground objects (in effect using cameras as occupancy sensors) and states that pixels corresponding to occupancies on a reference plane will consistently warp (under homographies of the reference plane) to foreground regions in every view. The reason we use foreground likelihood maps instead of binary foreground maps is to delay the thresholding step to the last possible stage. Starting from a reference scene plane, the homographic occupancy constraint is applied using multiple planes parallel to the reference plane to robustly localize scene objects. This added step significantly reduces false positives and missed detections due to artifacts like shadows, or when it cannot be guaranteed that a single plane will consistently be occupied by scene objects.

To track, we obtain object scene occupancies for a window of time and stack them together, creating a space-time volume. Occupancies belonging to the same person form contiguous spatio-temporal regions that are clustered using a graph cuts segmentation approach. This is achieved by designing an energy functional that combines scene occupancy information and spatio-temporal proximity. The

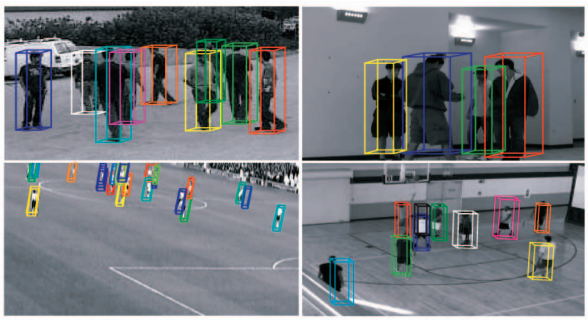

Fig. 1. Examples of cluttered and crowded scenes used to test our approach. For illustration purposes, only one view for each scene is shown.

energy functional is minimized over the spatio-temporal grid using graph cuts that result in the segmentation of contiguous spatio-temporal clusters. Each cluster is the track of a person and a slice in time of this cluster gives the tracked location.

We assume that at least one reference scene plane is visible in the views. This is a reasonable assumption in typical surveillance installations which are monitoring people in busy crowded places where usually the ground plane or a planar structure like a building wall is visible. Such planar structures usually occupy a large enough image region to be automatically detected and aligned using robust methods of locking onto the dominant planar motion. Homographies induced by the reference plane between views are computed using SIFT feature matches and employing the RANSAC algorithm. Homographies of planes parallel to the reference plane are obtained in the framework of plane-to-plane homologies, using the vanishing point of the direction normal to the reference plane. The result is that our approach is purely image based and performs fusion in the *image plane* without requiring to go in 3D space, and thus eliminating the need for fully calibrated cameras.

The rest of this paper is structured as follows: In Section 2, we discuss related work. Section 3 details the observation and theory behind the homographic occupancy constraint. In Section 4, we present our algorithm that uses the homographic occupancy constraint to localize people on multiple planes in the scene. Section 5 describes our tracking methodology. Section 6 details our experiments and results providing insight into the utility and efficiency of our method. We conclude this paper in Section 7.

2 Related Work

In this section, we provide context for the proposed approach in the backdrop of previous work. Broadly speaking, the literature on tracking multiple occluding targets in cluttered scenes can be divided into two categories: monocular approaches and multiview approaches, some of which are described below. For a detailed review of the state of the art in tracking research, the reader is referred to the recent survey by Yilmaz et al. [61].

2.1 Monocular Approaches

There is extensive literature on single-camera detection and tracking algorithms for multiple targets. This approach has the inherent advantage of simple and easy deployment, but has to rely on limited 3D information in a single view.

Blob tracking is a popular low-cost approach for tracking objects [17], [16]. It entails extracting blobs in each frame, and tracking is performed by associating blobs from one frame to the next. The $BraMBLe$ system [21], for example, is a multiblob tracker that generates a blob-likelihood based on a known background model and appearance models of the tracked people. Its performance degrades when multiple objects merge into one blob due to proximity or occlusions. Alternate approaches maintain explicit object states with position, appearance, and shape. Zhao and Nevatia [63], [64] present interesting results when tracking multiple people with a single camera. They use articulated ellipsoids to model human shape, color histograms to model different people's appearance, and an augmented Gaussian distribution to model the background for segmentation. Once moving head pixels are detected in the scene, a principled MCMC approach is used to maximize the posterior probability of a multiperson configuration. This concept of global trajectory optimization was previously explored in [30] and more recently in [2]. It also forms the basis of our tracking formulation; however, there is an important difference. Our approach utilizes fusion of multiple views at multiple scene planes and trajectory optimization on scene occupancy probabilistic data that combines the task of detection and tracking seamlessly.

Okuma et al. [40] propose a noteworthy combination of Adaboost for object detection and particle filters for multiple-object tracking. The combination of these two approaches leads to fewer failures than either one on its own, as well as addressing both detection and consistent track formation in the same framework. Brostow and Cipolla [5] present a probabilistic framework for the clustering of feature point trajectories to detect individual pedestrians in a crowd. These and other similar approaches like [28], [36], [47], [54] skip the modeling of articulations in favor of appearance models trained for specific unoccluded views of their respective subjects. As a result, they are challenged by fully and partially occluding objects, as well as appearance changes.

A number of monocular tracking techniques have been devised for handling occlusions. The typical approach is to detect the occurrence of occlusion by blob merger [17]. The methods for tracking feature points simply detect the occlusion of a feature point as the disappearance of the point being tracked [53]. In recent years, tracking techniques using object contours [62], [34] and appearances [59], [20], which represent and estimate occlusion relationships between objects by using hidden variables of depth ordering of objects toward the camera, have been proposed. Wu et al. [59] incorporate an additional hidden process for occlusion into a dynamic Bayesian network and rely on the statistical inference of the hidden process to reveal occlusion relations. Senior et al. [52] use appearance models to localize objects and use disputed pixels to resolve their depth ordering during occlusions. However, the system cannot maintain object identity after occlusions. Jojic and Frey [23] and Tao et al. [56] both model videos as a layered composition of objects and use EM to infer object's appearances and motions. Recently, Perera et al. [44] proposed a two-stage framework, which involves one-to-one correspondence followed by a split and merge analysis,

for linking tracks across occlusions. Most of the aforementioned approaches rely on partial observations, which makes it difficult to handle full occlusions. In addition, small and consistent motions are assumed to predict the motion patterns of objects through occluded views. This causes problems in dealing with long periods of occlusions of an object under unpredictable motions. In spite of the current body of knowledge, we believe monocular methods have limited ability to handle occlusions involving several objects, generally two or three, because the single viewpoint is intrinsically unable to observe the hidden areas.

2.2 Multicamera Approaches

The use of multiple cameras soon becomes necessary when one wishes to accurately detect and track multiple occluding people and compute their precise locations in a complex environment. Multiview tracking techniques intend to decrease the hidden regions and provide 3D information about the objects and the scene by making use of redundant information from different viewpoints.

In [25], Kelly et al. constructed a 3D environment model using the voxel feature. Humans were modeled as a collection of these voxels to resolve the camera-handoff problem. In [50], Sato et al. use CAD-based environment models to extract 3D locations of unknown moving objects. Jain and Wakimoto [22] also utilized calibrated cameras to obtain 3D locations of each object in an environment model for the Multiple Perspective Interactive Video. These works were characterized by the use of environment models and calibrated cameras. Multitarget tracking by association across multiple views was addressed in a series of papers from the latter half of the 1990s. In [38], Nakazawa et al. constructed a state transition map that linked regions observed by one or more cameras, along with a number of action rules to consolidate information between cameras. Orwell et al. [41] present a tracking algorithm to track multiple objects in multiple views using "color" tracking. They model the connected blobs obtained from background subtraction using color histogram techniques and use them to match and track objects. Cai and Aggarwal [6] extend a single-camera tracking system by starting with tracking in a single camera view and switching to another camera when the system predicts that the current camera will no longer have a good view of the subject. Spatial matching was based on the euclidean distance of a point to its corresponding epipolar line. In [24], individuals are tracked both in image planes and top view using a combination of appearance and motion models. Bayesian networks were used in several papers as well. In [7], Chang and Gong used Bayesian networks to combine geometry (epipolar geometry, homographies, and landmarks) and recognition (height and appearance) based modalities to match objects across multiple sequences. Bayesian networks were also used by Dockstader and Tekalp in [10], to track objects and resolve occlusions across multiple calibrated cameras. Integration of stereo pairs is another popular approach, adopted by [31], [37], [9], [1] among others. Krumm et al. [31] use stereo cameras and combine information from multiple stereo cameras in 3D space. They perform background subtraction and then detect human-shaped blobs in 3D space. Color histograms are created for each person and are used to identify and track people. Mittal and Larry [37] use a similar method to combine information in pairs of stereo cameras. Regions in different views are compared with each other, and back projection in 3D space is done in a manner that yields 3D points guaranteed to lie inside the objects.

Although these methods attempt to resolve occlusions, the underlying problem of using features (appearance templates, blob shapes) that might be corrupted due to occlusions remains. Second, occlusion reasoning in these approaches is typically based on temporal consistency in terms of a motion model, whether it is Kalman filtering or more general Markov models. As a result, these approaches may not be able to recover if the process begins to diverge. The scenes shown in Fig. 1 would be difficult to resolve for the majority of these methods. As well as cases of near total occlusion, the people are dressed in very similar colors. Using blob shapes or color distributions for region matching across cameras may lead to incorrect segmentations and detections.

The homographic occupancy constraint [27] presented in this paper fuses information from multiple views using sound geometrical constructs and resolves occlusions by localizing people on multiple scene planes. We essentially attempt to find image locations of scene points that are guaranteed to be occupied by people. These occupancies are then used to resolve occlusions and track multiple people. In this context, the work by Mittal and Larry [37], Franco and Boyer [12], Berclaz et al. [2], Yang et al. [60], and the parallel work on range sensor-based occupancy grids for robot navigation is quite relevant [11], [57]. However, unlike these approaches, which fuse information in 3D space requiring calibrated cameras, our approach is completely image based and requires only 2D constructs like planar homographies to perform fusion *in the image plane* without requiring to go in 3D space.

Alternative approaches to homography-based tracking by Kalman and particle filtering were presented in [19] and [29], respectively. The authors in [19] extracted the principal axes of upright humans tracked in each view and then combined multiple views using planar homographies. Homography-based 2D segmentation and tracking of objects has also been studied in the intelligent transportation domain, for instance, the recent work by Park and Trivedi [42], [43]. They propose to combine multiple view data, which is then augmented with contextual domain knowledge for the analysis and query of person-vehicle interactions for situational awareness and pedestrian safety. In [26], Khan and Shah proposed an approach that avoided explicit calibration of cameras and instead utilized constraints on the field of view (FOV) lines between cameras, learned during a training phase, to track objects across the cameras. These and similar techniques track objects in individual uncalibrated views and then create associations across views for better localization; this approach declines with increasing densities of scene objects. We neither localize nor track people from any single camera, or camera pair; rather, evidence is gathered from all the cameras into a unified synergistic framework where occlusion resolution, detection, and tracking are performed simultaneously. The detection and tracking results are then propagated back to each view.

3 Homographic Occupancy Constraint

Consider a scene containing a reference plane being viewed by a set of wide-baseline stationary cameras. The background models in each view are available and, when an object appears in the scene, it can be detected as foreground in each view using background difference. Any scene point lying inside the foreground object in the scene will be projected to a foreground pixel in every view. This also applies for scene points inside the object that lie on the reference plane except, however, that the projected image locations in each view will be related by homographies induced by the reference plane. We can state the following:

Proposition 1. *If $\exists P \in \mathbf{R}^3$ such that it lies on scene plane π and is inside the volume of a foreground object, then the image projections of the scene point P given by $p_1, p_2, \ldots, p_n$ in any n views satisfy both of the following:*

- *$\forall_i$, if Ψ_i is the foreground region in view i, then $p_i \in \Psi_i$.*
- *$\forall_{i,j} p_i = [H_{i_\pi j}]p_j$, where $H_{i_\pi j}$ is the homography induced by plane π from view j to view i.*

Warping a pixel from one image to another using the homography induced by a reference scene plane amounts to projecting a ray through the pixel onto the piercing point (point where the ray intersects the reference plane) and then projecting it to the second camera center. If the pixel's piercing point is inside (occupied by) a foreground object in the scene, it follows from Proposition 1 that the pixel will warp to foreground regions in all views. This can be formally stated as follows:

Proposition 2. *Let Φ be the set of all pixels in a reference view r and let $H_{i_\pi r}$ be the homography of plane π in the scene from the reference view to view i. If $\exists p \in \Phi$ such that the piercing point of p with respect to π lies inside the volume of a foreground object in the scene, then $\forall_i p'_i \in \Psi_i$, where $p'_i = [H_{i_\pi r}]p$ and Ψ_i is the foreground region in view i.*

We call Proposition 2 the *homographic occupancy constraint* [27]. Notice that this does not distinguish between foregrounds in different views that may correspond to different objects. It is essentially using camera sensors as scene occupancy detectors with foreground interpreted as occupancy in the line of sight of the image sensor. Although the foreground regions associated across views may correspond to different scene objects (specifically the nearest foreground object in the line of sight of the particular image sensor), the homographic occupancy constraint insures that they all correspond to the same scene occupancy.

This has the dual action of localizing people in the scene as well as resolving occlusion, which is described in Fig. 2. Fig. 2a shows a scene containing a person viewed by a set of cameras. The foreground regions in each view are shown as white on a black background. A pixel which is the image of the feet of the person will have a piercing point on the ground plane (the reference plane for this example) that is inside the volume of the person. According to the homographic occupancy constraint, such a pixel will be warped to foreground regions in all views. This is demonstrated by the pixel in view 1 of Fig. 2a that has a blue ray projected through it. Foreground pixels that do not satisfy the homographic occupancy constraint are images of points off the ground plane. Due to plane parallax, they are warped to background regions in other views. This is demonstrated by the pixel with a red ray projected through it. Fig. 2b shows how the homographic occupancy constraint would resolve occlusions. The blue person is occluding the green person in view 1. This is apparent by the merging of their foreground blobs. In such a case, there will be two sets of pixels in view 1 that satisfy the homography constraint. The first set will contain pixels that are image locations of blue person's feet (same as in Fig. 2a). The other set of pixels is those that correspond to the blue person's torso region, but are occluding the feet of the green person. Even though these pixels are image locations of points off the ground plane, they have piercing points inside a foreground object, which in this case is the green person. This process creates a seemingly translucent effect detecting feet regions even if they are completely occluded by other people. Clearly, having more people between the blue and the green person will not affect the localization of the green person on the ground plane.

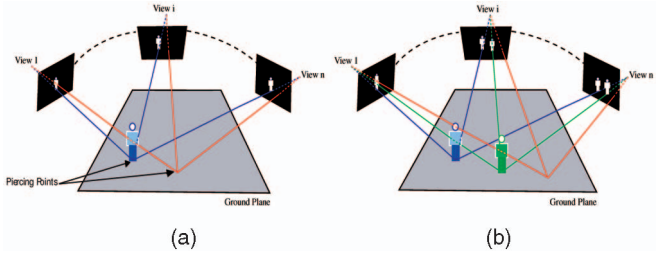

Fig. 2. The figure shows people viewed by a set of cameras. The views show the foreground detected in each view. For (a), the blue ray shows how the pixels that satisfy the homographic occupancy constraint warp correctly to foreground in each view, while others have plane parallax and warp to background. (b) demonstrates how occlusion is resolved in view 1. Foreground pixels that belong to the blue person but are occluding the feet region of the green person satisfy the homographic occupancy constraint (the green ray). This seemingly creates a see-through effect in view 1, where the feet of the occluded person can be detected.

It should be noted that the homographic occupancy constraint is not limited to the ground plane and, depending on the application, any plane in the scene could be used. In the context of localizing people in a surveillance scenario, the ground plane is typically a good choice if it is clearly visible. In other scenarios, a building wall or any planar landmark can be used as the reference plane. In the next section, we develop an operator that uses this approach to localize people on a reference plane.

4 Localizing People

Let $\Phi_1, \Phi_2, \ldots, \Phi_n$ be the images of the scene obtained from n uncalibrated cameras. Let Φ_r be a reference view. $H_{i_\pi r}$ is homography of the reference plane π between the reference view and any other view i. Using homography $H_{i_\pi r}$, a pixel p in the reference image is warped to pixel p'_i in image Φ_i. Let $x_1, x_2, \ldots, x_n$ be the observations in images $\Phi_1, \Phi_2, \ldots, \Phi_n$ at locations $p'_1, p'_2, \ldots, p'_n$, respectively, (i.e., $x_i = \Phi_i(p'_i)$). Let X be the event that pixel p has a piercing point inside a foreground object (i.e., p represents the reference plane π location of a foreground object in the scene). Given $x_1, x_2, \ldots, x_n$, we are interested in finding the probability of event X happening, i.e., $P(X|x_1, x_2, \ldots, x_n)$.

Using Bayes law

$$P(X|x_1, x_2, \ldots, x_n) \propto P(x_1, x_2, \ldots, x_n|X)P(X). \quad (1)$$

The first term on the right-hand side of (1) is the likelihood of making observation $x_1, x_2, \ldots, x_n$, given event X happens. By conditional independence, we can write this term as

$$P(x_1, x_2, \ldots, x_n|X) = P(x_1|X) \times P(x_2|X) \times \ldots \times P(x_n|X). \quad (2)$$

Now, the homographic occupancy constraint states that if a pixel has a piercing point inside a foreground object, then it will warp to foreground regions in every view. Therefore, it follows that

$$P(x_i|X) \propto L(x_i), \quad (3)$$

where $L(x_i)$ is the likelihood of observation x_i belonging to the foreground. Plugging (3) into (2) and back into (1), we get

$$P(X|x_1, x_2, \ldots, x_n) \propto \prod_{i=1}^{n} L(x_i). \quad (4)$$

The value of $P(X|x_1, x_2, \ldots, x_n)$ given by (4) represents the likelihood of the scene location being occupied by the foreground object. In effect, we are hypothesizing in the reverse direction by reasoning about scene occupancies from the fusion of scene observations.

4.1 Modeling Clutter and FOV Constraints

So far, we have assumed the scene point under examination is inside the FOV of each camera, limiting our analysis to overlapping region of the multiview setup. Also, the fusion operation in (4) assigns uniform prior precedence to each view. Due to the varying amounts of clutter in a particular view, the degree of confidence in foreground detection will be effected. Clutter may cause false detections or miss the foreground in some cases. Therefore, in this section, we propose to use a measure of clutter, in order to weigh the foreground likelihood information detected from different views in our fusion model.

Schmieder and Weathersby [51] proposed the concept of an RMS clutter metric of the spatial-intensity properties of the scene. Due to its robustness and applicability, it is one of the most commonly used clutter measures. Experimental results that have been reported in the literature [51], [49] show a high correlation between the average target detection time by human subjects and the Schmieder and Weathersby (SW) clutter metric. The SW clutter metric is computed by averaging the variance of contiguous square cells over the whole image:

$$C = \sqrt{\frac{1}{N}\sum_{k=1}^{N} \sigma_k^2}, \quad (5)$$

where σ_k^2 is the variance of pixel values within the kth cell, and N is the number of cells or blocks the picture has been divided into. Typically, N is defined to be twice the length of the largest target (in our case, humans) dimension. We compute the clutter metric for each view at each time instant on the *foreground likelihood maps* obtained from background modeling (for each pixel, the likelihood of being foreground); therefore, σ_k^2 in (5) is the variance of foreground likelihood values in the kth cell. Fig. 3 shows some of the views of data sets used in our experiments (first row), their corresponding foreground likelihood maps (second row), and the SW clutter values obtained from them. As illustrated by Fig. 3, the views with more noise and clutter in the foreground likelihood maps have a greater SW clutter metric value.

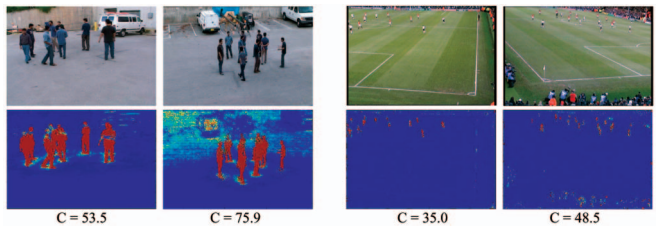

Fig. 3. The first row shows images from two of our test sequences (two views each). The second row shows foreground likelihood maps for views in the first row, where redder corresponds to greater foreground likelihood. The SW clutter metric is computed on these foreground likelihood maps. It can be visually corroborated that views with noisy foreground likelihood maps have higher clutter value.

In order to assign higher confidence to foreground detected from views with lesser clutter, we use the following method. For each foreground likelihood map i, we use clutter C_i, computed using (5) as its prior weight in the log likelihood of the fusion operation in (4):

$$\log(P(X|x_1, x_2, \ldots, x_n)) \propto \sum_{i=1}^{n} \frac{1}{\tau C_i} \log(L(x_i)), \quad (6)$$

where $\tau = \sum_i \frac{1}{C_i}$ is a normalizing factor. The effect of modeling clutter on the performance of our approach is further discussed in the results and experiments section.

Though (6) ensures that the evidence from all available views is combined to maximize the certainty in the localization hypothesis, it also assumes that the region of space under analysis is inside the overlapping FOV of all cameras. If a scene point is outside the FOV of one or more cameras, the missed detection causes the remaining, possibly correct detections from other views to be discarded. This obvious problem is corrected by modifying the fusion operator as

$$\log(P(X|x_1, x_2, \ldots, x_n)) \propto (\Sigma_i \delta(x_i)) \sum_{i=1}^{n} \frac{1}{\tau C_i} \Gamma(x_i), \quad (7)$$

where

$$\delta(x) = \begin{cases} 1 & \text{if } x \text{ inside image dimensions;} \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\Gamma(x) = \begin{cases} \log(L(x)) & \text{if } \delta(x) = 1; \\ 0 & \text{otherwise.} \end{cases}$$

The form of (7) ensures that if a scene point is outside the FOV of a camera, that particular view will not effect the fusion results. Also, the normalizing term $\Sigma_i \delta(x_i)$ guarantees higher confidence in regions with greater view overlap.

A pixel p in a reference view can be classified as an image of the reference scene plane localization of an object if the occupancy likelihood given by (7) is above a threshold. In the case foreground objects are people and the reference scene plane is the ground plane, pixel p will correspond to

the feet of a person in the scene. Since pixel p and its warped locations in other views $p'_1, p'_2, \ldots, p'_n$ all have the same piercing point, they all correspond to the same location on the reference scene plane. Therefore, by finding p in the reference view that satisfies the homographic occupancy constraint, we have in fact, localized the particular person in all views (i.e., $p'_1, p'_2, \ldots, p'_n$). This strategy implicitly resolves the issue of correspondences across views and makes the choice of reference view irrelevant (chosen arbitrarily).

4.2 Localization at Multiple Planes

As pointed out earlier, the homographic occupancy constraint is not limited to any single plane in the scene. In fact, fusion can be performed on multiple planes to increase robustness of the localization. There might be cases where occupancy on the scene reference plane is intermittent, for example, when the ground plane is used and the people are running or jumping, resulting in minimal contact with the ground. Another example of this is when, in the absence of a visible ground plane, a building back wall is used where there is actually no occupancy requiring the use of planes parallel to the back wall plane. Our approach is, therefore, to perform localization at the reference plane, as well as multiple *imaginary* planes parallel to the reference plane along the normal direction. Evidently, when obtaining homographies induced by the imaginary planes, a conventional feature correspondence-based approach is not feasible. However, as we have shown (in the Appendix), if we have the homography $H_{i_\pi j}$ induced by a reference scene plane π between views i and j, then the homography $H_{i_\phi j}$ induced by a plane ϕ parallel to π is given by

$$H_{i_\phi j} = \left(H_{i_\pi j} + [0|\gamma \mathbf{v}_{ref}]\right)\left(I_{3\times 3} - \frac{1}{1+\gamma}[0|\gamma \mathbf{v}_{ref}]\right), \quad (8)$$

where $\mathbf{v}_{ref}$ is the vanishing point of the normal direction and γ is a scalar multiple controlling the distance between the parallel planes.

In our implementation, we typically use the ground plane as the reference scene plane and the up direction as the reference direction. The reference plane homographies between views were automatically calculated with SIFT [32] feature matches and using the RANSAC algorithm [18]. Vanishing points for the reference direction were computed by detecting vertical line segments in the scene and finding their intersection in a RANSAC framework as in [48]. In the absence of robustly detectable vertical landmarks, we estimate the vanishing point from observations of walking people similar to the approach described in [33]. It should be noted that the particular values of γ are not significant, we are only interested in the range of γ for planes that span the body of the object (e.g., if the object is a person, then starting from the ground plane to a plane that is parallel to the ground plane and touching the tip of the head). The computation of this range for γ is quite straightforward since, outside this range, the homographic occupancy constraint is not satisfied (i.e., the occupancy likelihood approaches zero). In the next section, we outline our algorithm for using the homographic occupancy constraint to obtain people localization likelihoods on multiple planes.

4.3 Localization Algorithm

Our algorithm for localizing people is rather simple. First, we obtain the foreground likelihood maps in each view. This is done by statistically modeling the background using a Gaussian distribution [55], [58] and finding the probability for each pixel belonging to the foreground. In the second step, instead of warping every pixel in the reference image to every other view, we perform the equivalent step of warping the foreground likelihood maps from all the other views on to the reference view. These warped foreground likelihood maps are then fused according to (7), to produce what we call a "synergy map" of the reference plane [27]. The synergy map is a 2D grid of object occupancy likelihoods. The process is repeated on multiple planes parallel to the reference plane to obtain a series of synergy maps. Though a threshold can then be applied to each synergy map to obtain reference plane localizations, this would require the estimation of an optimal threshold at each fusion plane. Besides, there are interdependencies between occupancies at different planes. We therefore delay the act of thresholding and feed the soft occupancy likelihood information from the synergy maps directly into our tracking module, the details of which are presented in Section 5. The following are the steps in our localization algorithm:

Objective Localize people on N planes parallel to scene reference plane

Localization Algorithm

1) Obtain the foreground likelihood maps $\Psi_1, \Psi_2 \ldots, \Psi_n$.
   - Model Background using a Mixture of Gaussians.
   - Perform Background Subtraction to obtain foreground likelihood information.
2) Obtain reference plane homographies and vanishing point of reference direction.
3) **for** $i = 1$ to N
   - Update reference plane homographies using (8)
   - Warp foreground likelihood maps to a reference view using homographies of the reference plane.
     - Warped Foreground Likelihood maps: $\Psi'_1, \Psi'_2 \ldots, \Psi'_n$
   - Fuse $\Psi'_1, \Psi'_2 \ldots, \Psi'_n$ at each pixel location p of the reference view according to (7) to obtain synergy map θ_i
   - end **for**
4) Arrange θ_is as a 3D stack in the reference direction
   - $\Theta = [\theta_1; \theta_2 \ldots \theta_n]$

Fig. 4 shows the algorithm applied to one of our test scenes. The first two rows of Fig. 4a show the foreground likelihood information in the available views. View 4 was chosen as the reference view and the other views were warped to the reference view with the homography of the reference scene plane (the ground plane). The synergy map from the fusion is shown in the third row and it clearly highlights the feet regions of the people. Notice how occlusions are resolved and the ground locations of people

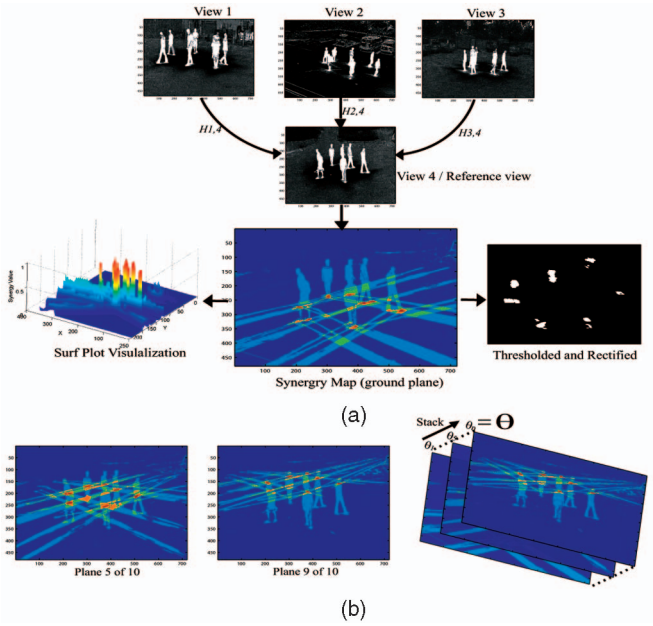

Fig. 4. (a) The first two rows show the foreground likelihood maps obtained from the background model on the available views. The color map assigns a brighter palette to higher values. View 4 was chosen as the reference view. The third row shows the synergy map obtained by warping views 1, 2, and 3 onto view 4 and fusing them together. A surf plot of the synergy plot is shown on the left. The ground locations of the people stand out clearly. For the sake of visualization, we show the binary image on the right, obtained by applying a threshold and rectifying with the ground plane. (b) On the left are two more synergy maps using planes parallel to the ground plane (planes 5 and 9 of 10). On the right is the illustration of the synergy maps θ_is obtained from all fusion planes being packaged up in a single 3D data structure Θ that is passed on to the tracking module.

are detected. Fig. 4b shows the synergy maps produced by fusing at multiple planes parallel to the ground plane. For the purpose of tracking, the synergy maps were rectified with the reference planes. The rectified image is an accurate picture of the relative ground locations of the people in the scene. The 2D synergy maps are then stacked together into a 3D grid Θ of object occupancy likelihoods that represents a discrete sampling (on the reference planes) of the continuous scene occupancy space. As described in the next section, tracking is performed using Θs on all synergy maps simultaneously.

5 Tracking

Our tracking methodology is based on the concept of spatio-temporal coherency of scene occupancies created by objects. Assuming that a particular scene location at a specific time can be occupied by only a single individual, we hypothesize that over time spatially coherent scene occupancies correspond to the tracks of scene objects. We therefore propose a look-ahead technique to solve the tracking problem by using a sliding window over multiple frames. This information gathering over time, of systems simulating the cognitive processes, is supported by many researchers in both computer vision and psychology (e.g., [15], [35], [39]). Neisser [39] propose a model in which the perceptual processes continually interact with the incoming information to verify hypotheses formed on the basis of available information up to a given time instant. Marr's principle of least commitment [35] states that any inference in a cognitive process must be delayed as much as possible. Many existing algorithms use similar look-ahead strategies or information gathering over longer intervals of time (for example, by backtracking) [45], [46].

Let us denote by $\boldsymbol{\xi}^{\mathbf{n}} = (\xi_1^n, \xi_2^n, \ldots, \xi_t^n)$ the trajectory of spatio-temporal occupancies by individual n, where ξ_i^n represents the spatial localization of the individual n in the occupancy likelihood information Θ_i at time i. Given the occupancy likelihood information from our localization algorithm for a sliding time window of t frames $\Theta_1, \Theta_2, \ldots, \Theta_t$, the tracks are obtained by maximizing the posterior conditional probability:

$$[\hat{\boldsymbol{\xi}}^{\mathbf{1}}, \ldots, \hat{\boldsymbol{\xi}}^{\mathbf{n}}] = \arg\max_{l_1,\ldots,l_n} P(\boldsymbol{\xi}^{\mathbf{1}} = l_1, \ldots, \boldsymbol{\xi}^{\mathbf{n}} = l_n | \Theta_1, \Theta_2, \ldots, \Theta_t). \tag{9}$$

To achieve this, we define an energy function that combines occupancy regularization and region information, in a fashion similar to Mumford-Shah style functions. The global minimum is found by using graph cut techniques that will be discussed next.

5.1 Graph Cuts Trajectory Segmentation

For a time window of t frames, we obtain the scene occupancy likelihood information from our localization algorithm: $\Theta_1, \Theta_2, \ldots, \Theta_t$. Each Θ_i is a 3D grid of object occupancy likelihoods, obtained from multiview fusion at multiple scene planes as described in previous sections. By arranging Θ_is in the time dimension, we create what we call a *spatio-temporal occupancy likelihood grid*: $\hat{\boldsymbol{\Theta}} = [\Theta_1; \Theta_2; \ldots; \Theta_t]$. Each location or node in this 4D grid contains the object presence likelihood for a specific space-time point. Our goal is to segment $\hat{\boldsymbol{\Theta}}$ into background (nonoccupancies) and object occupancy trajectories with the following criteria:

1. Grid locations with high occupancy likelihoods have a higher chance of being included in object trajectories.
2. Object trajectories are spatially and temporally coherent.

Given these criteria, we define our energy function as

$$\mathcal{E} = \mu \sum_{p \in \mathcal{P}} -\hat{\boldsymbol{\Theta}}(p) + \sum_{(p,q) \in N} B_{p,q}, \tag{10}$$

where $\mathcal{P}$ is the set of all grid locations/nodes in $\hat{\boldsymbol{\Theta}}$, N is the set of grid locations in a neighborhood, and $B_{p,q} \propto e^{-dist(p,q)/2\tau^2}$, where $dist(p,q)$ is the 4D euclidean distance between grid locations p and q and τ is a normalizing factor. The first term in (10), also known as the data term, imposes scene occupancy. The second term known as the smoothness term imposes the constraint of spatio-temporal coherency. By minimizing (10), the idea is to obtain regions in the spatio-temporal occupancy likelihood space that have high presence probabilities (small negative log likelihood) and are smooth, meaning that they are close to each other both in space and time.

In order to minimize the energy function given in (10), we use graph cut techniques. Graph cuts have been used in the past in the context of tracking humans [13] but, unlike other approaches designed to work in the appearance domain, our formulation is purely in the scene occupancy domain. Our undirected graph $G = (V, E)$ is as follows: The

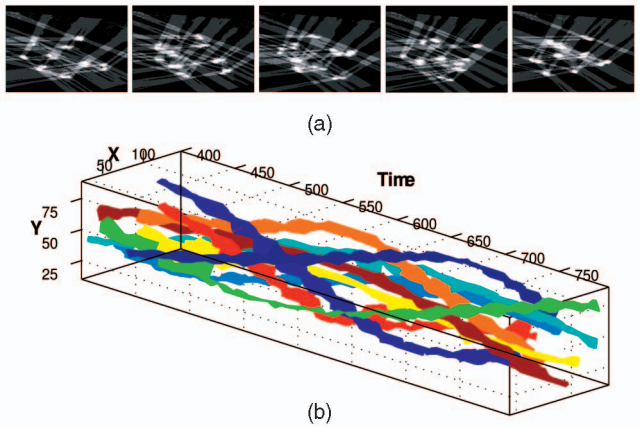

Fig. 5. (a) A sequence of synergy maps at the ground reference plane of nine people obtained using our algorithm. In (b), we show the XY cut (corresponding to the ground reference plane) of the 4D spatio-temporal occupancy tracks obtained using our tracking approach. Different tracks are colored differently to help in visualization. The spiraling pattern of the racks is only a coincidence. This occurred because the people were walking in circles in this particular sequence.

set of vertices is the set of spatio-temporal grid locations augmented by the source S and sink T vertices: $V = \mathcal{P} \cup S, T$. The set of edges consists of all neighboring pairs of nodes, as well as edges between each node and the source and sink: $E = \mathcal{N} \cup \{(p, S), (p, T) : p \in \mathcal{P}\}$. In terms of the weights on the edges, there are three cases to consider. If $(p, q) \in \mathcal{N}$, then $w(p, q) = B_{p,q}$. On the other hand, if the edge contains the source S or sink T as one of its vertices, then $w(p, \{S, T\}) = -\hat{\Theta}(p)$.

It is relatively straightforward to show that the minimum cut on the graph G corresponds to the minimum values of the energy function equation (10) [3]. The specific algorithm we use is the α-expansion algorithm described in [4]. To keep the problem computationally tractable, we quantized the XY plane as a 100 $\times$ 100 grid. The number of fusion planes, the quantization on the Z direction, was between 10 and 20, depending on the experiment. The sliding time window size was kept at 15 frames for each experiment (which corresponds to 1 second in the real world of a 15 fps video). The sliding window has minimal overlap, one frame, e.g., the last frame of window_i and the first frame of window_{i+1} are the same. The overlap is used to pass on the track identities. The identities are initialized in the first window. For successive windows, each segmented track is given the ID that a previous window assigned it at the overlap frame. Though larger window size and greater overlap can be used to improve performance, we found the improvement was not significant enough to justify the increased load of processing.

Note that we do not make hard detection decisions and use them for tracking. Rather, tracking and detection are intimately tied together and are performed simultaneously when we segment out space-time tracks from the occupancy likelihood data $\hat{\Theta}$. This, we believe, is an elegant solution to the inherently coupled tasks of detection and tracking. The advantage of this approach is twofold. First, false negatives and false positives are reduced compared with a traditional threshold-based detection (see experimental evaluation). In cases where a missed detection from thresholding (for one or more frames) would cause a track

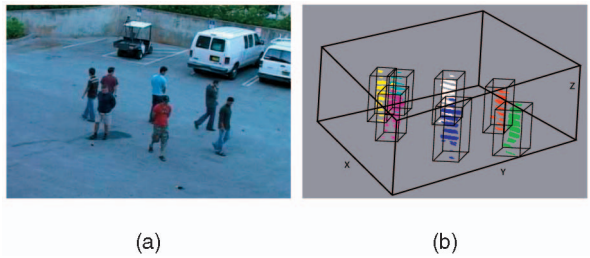

Fig. 6. (a) The reference view. (b) The 3D marginal of the 4D spatio-temporal occupancy tracks at a particular time. Notice the gaps in localizations for each person's color-coded regions. This is because only 10 planes parallel to the ground in the up (Z) direction were used.

to be lost, we are able to recover the tracks and, hence, the detections. This is because the energy functional minimized with graph cuts combines both occupancy probability and spatio-temporal smoothness. For instance, if the occupancy probability for a person does not pass the detection threshold for a particular frame in the time window, our track segmentation approach still includes the region in the particular frame to reduce the cost incurred by having neighboring nodes in the space-time occupancy grid more than one frame away (smoothness). Second, this approach helps in cases where a thresholded detection results in artifacts in a single person's detection, i.e., the region is split into two or more very close but unconnected regions. Such regions are typically merged together by our approach. This property of our approach may also cause tracks of two or more people to merge if they come very close to each other; however, these are uncommon cases and resolved in the long run, since people's body parts tend to remain closer to them than to other people. In situations where detection results using a thresholding approach are sufficiently good (typically not the case in challenging scenarios as demonstrated in our results section), a simple tracker like EKF or the more sophisticated particle filtering tracking can be used as has been attempted in past literature [63], [28]. However, such trackers do not naturally handle splits and merges, and require an explicit split-merge analysis separate from the tracking. This, of course, is naturally handled in our track segmentation approach.

In Fig. 5b, we describe an example of the spatio-temporal occupancy tracks obtained for a scene containing nine people, which was used in one of our experiments. The figure shows the results after processing multiple time windows. Only the 3D marginal on the ground reference plane of the actual 4D occupancy tracks are shown. A slice in time is the tracked location of people in the scene. Fig. 6 shows the 3D marginal view of the 4D spatio-temporal occupancy tracks at a particular time instant for one of our experiments. Track windows are plotted around each track. The tracking results are propagated to other views using the inverse of the homographies used in localization.

6 Results and Discussions

In order to evaluate our approach, we conducted several experiments with both the data that we collected and multiview data sets which are publicly available. The current implementation of our localization algorithm runs on a Nvidia GeForce 7300 GPU and is capable of fusing up to eight views (480 $\times$ 720) on 10 fusion planes at the rate of

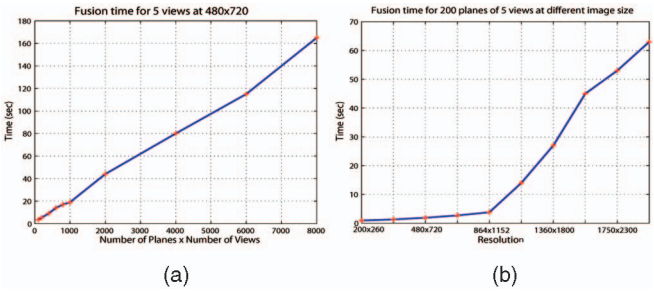

Fig. 7. Localization algorithm runtime on a GPU. (a) The execution time is linear with both the number of views and number of fusion planes. (b) Execution time with varying image resolution. As the resolution increases beyond the cache limit of the GPU, the performance drops.

approximately 35 fps. The computation cost increases linearly with both number of views and the number of planes at which fusion is performed. Fig. 7a shows the plot of run time against the product of the number of views and the number of planes on which localization was performed. The 2D nature of our localization algorithm has allowed us to harness the strength of the graphics card hardware acceleration. Fig. 7b shows the effect of increasing image resolution on the computation time. Notice the jump at 1,160 $\times$ 1,500, this is because, at such high resolutions, the GPU cache runs out of memory, resulting in reduced efficiency. In practice trials, the highest resolution images we used were 576 $\times$ 720. For the graph cuts-based tracking, we used a publicly available implementation, http://www.cs.cornell.edu/~rdz/graphcuts.html. On a 2.4 GHz core 2 duo machine, the implementation takes approximately 14-15 seconds on a grid of 100 $\times$ 100 $\times$ 10 $\times$ 15.

6.1 Video Data sets

We are reporting results on four multiview data sets. Three of these are novel challenging data sets that we captured ourselves. The fourth is a publicly available multicamera data set containing video sequences of a soccer match. The frame rate of all the sequences is 15 fps (subsampled from 30 fps streams). We computed error rates for these sequences in terms of false detections and missed detections or their sum which we call *detection error*. We define a true positive as the case when the bounding box of a tracked person (localization of a person over all fusion planes) encapsulates that particular individual in all views. For some of these sequences, we calculated detailed tracking errors by comparing them with the ground truth (manually marked tracking of the people).

Parking lot data set. This scene was captured using a video-surveillance dedicated setup of four synchronized cameras in a parking lot. The cameras were mounted at various heights ranging from 2 to 6 m and arranged unevenly in a rough circle. The sequence is over 3,000 frames and contains between five and nine people. The people were constrained to move in an area of approximately 5 $\times$ 5 m to simulate dense crowds and severe occlusions. We were attempting to increase the density of people and vary the number of views in order to study the breakdown thresholds and other characteristics. The homographic occupancy constraint was applied on 10 planes including and parallel to the ground plane in the up direction.

Fig. 8 shows our tracking results on the parking lot data set. For better visualization, only 2D track bounding boxes are plotted for this sequence. Due to the density of the gathering, occlusions were abundant and quite severe. An interesting thing to note is the color similarity of the people in the scene. A method that uses appearance (color distribution) matching across views would perform poorly in such a situation, whereas our method performs quite well. The top row of Fig. 8 shows a visualization of the top view, with configuration of the cameras overlaid. Camera overlap is color coded so that brighter yellow corresponds to higher overlap. The blue and red squares in the top view depict the true and false positives, respectively.

In Fig. 9, we show the quantitative and qualitative analysis of our results on this sequence. We analyzed the accuracy of our tracking results by comparing them with the ground truth, which was obtained by manually clicking the head and feet location of each person in each view. Our tracking accuracy measure was the perpendicular distance between the central axis passing through the localizations of a person in a particular view (the least square line through the centroids of localizations on all fusion planes) and the central axis obtained from manual marking (the ground truth line connecting head and feet). For this sequence, we had access to metric calibration data in order to convert image distances to actual world distances in terms of inches. The error was calculated for each tracked person in each view and was averaged over the number of people and the number of views. False positives and negatives were *not* included in the calculation of this measure. We call this the *total average track error*. Fig. 9a is the plot of the total average track error, computed at intervals of 100 frames. We varied the number of views by selecting a subset of the available views, in order to study the effect of reducing views on our approach. As expected, the total average track error significantly increased, from a mean of approximately 4 inches with four views (green plot) to over 14 inches with two views (red plot). The magenta plot in the figure shows the track error with four views, if clutter modeling is not used. As shown, the accuracy of tracking decreases if clutter modeling is not done. Clutter modeling helped in making the tracks more streamlined and precise by effectively pruning out false occupancy information in the periphery of detections. Also, in some cases (frames 550, 600, 800), the track error with three views and clutter modeling (blue plot) is close to that of four views without clutter modeling. Although we do not expect this particular trend to be the general pattern, it does indicate that modeling clutter has a useful impact. Detection error, on the other hand, was relatively unaffected with the use of clutter modeling, e.g., false detections arising due to scene locations being occluded from every view are not affected by using clutter modeling.

In our opinion, the other most significant factor influencing the performance of our algorithm is the density of the crowd or gathering. The greater the density, the more scene occupancies per unit area and, therefore, greater occlusions from vantage points resulting in difficulty with detection and localization. In Fig. 9b, we show three plots depicting a correlation between the density of people in the scene and the resulting detection error (sum of false positives and missed detections). To obtain people density, we calculate the area of the convex hull (in square feet, recall that we have metric calibration) containing the ground plane localizations of all tracked people at a given time instant

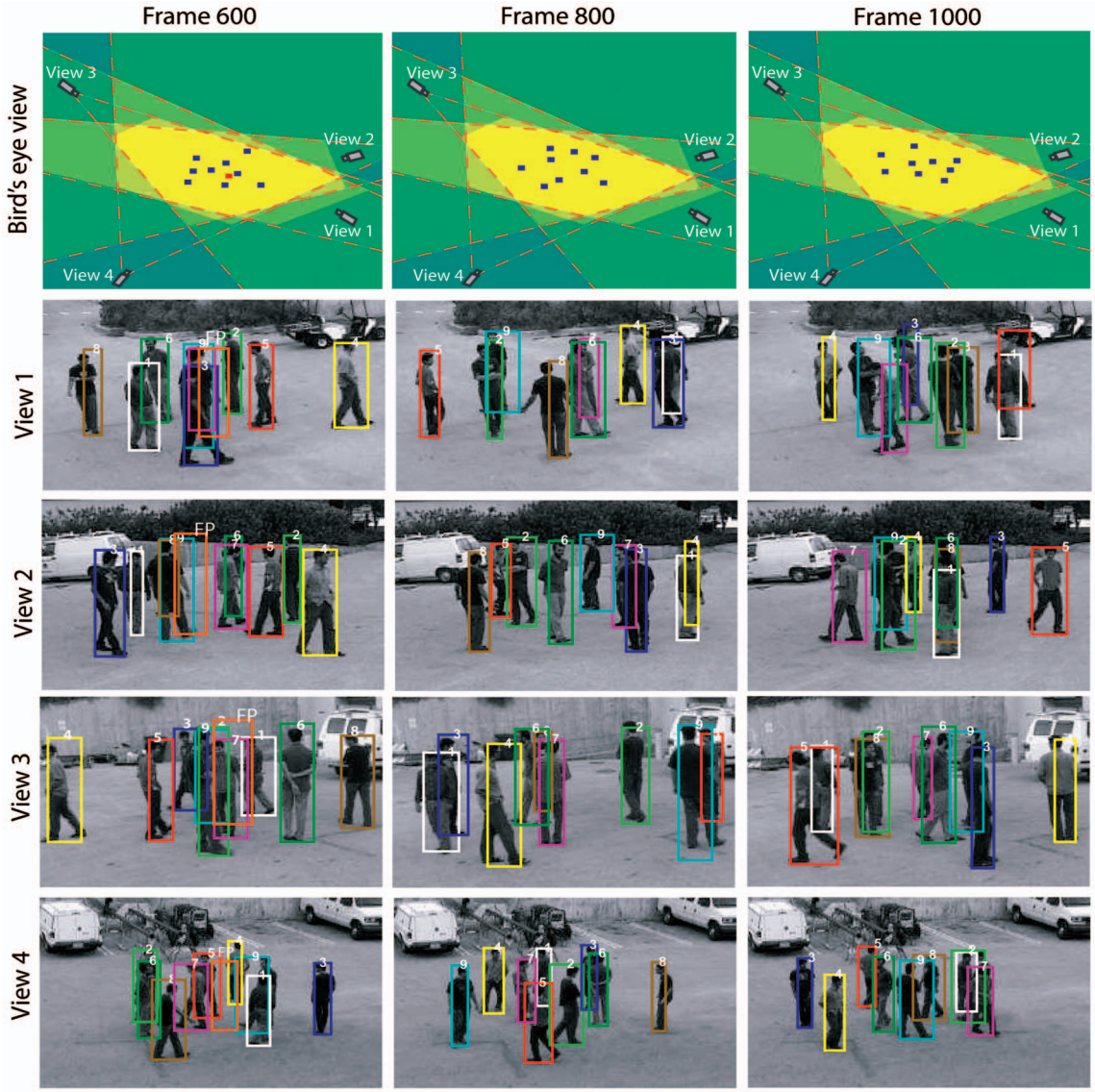

Fig. 8. *Parking lot sequence*. Tracking results for a scene containing nine people captured from four view points. The first row shows a visualization of the top view. It shows the camera FOV overlap, with higher overlap corresponding to yellower regions. Detection true and false positives are shown with blue and red squares, respectively. Rows 2-5 show the four camera views of the scene. Left to right, the columns correspond to frames 600, 800, and 1,000 in the respective views. In the camera views, for better visualization and less clutter, only 2D track bounding boxes are used unlike in Figs. 1, 11, 13, and 14. Track bounding boxes are color coded and numbered to show the correspondences that our algorithm accurately maintains across views. Notice the severe and recurring occlusions.

and divided the number of people by this value. The people density varied from 0.09 to 0.17 persons/sq ft as can be seen in the bottom plot of Fig. 9b. The top plot shows the detection error versus time at intervals of 100 frames. Notice the correlation, especially at the peaks of people densities. The correlation coefficient between the density plot and detection error plot is 0.7 for four views, 0.75 for three views, and 0.62 for two views.

Although we acknowledge that the number of views and people density are not the only scene factors influencing the performance of our approach, we believe that these are the most crucial. A detailed analysis can be quite exhaustive and can include camera configuration, relative people configuration (certain formations of people can occlude scene regions from all cameras) and scene geometry. These and other factors are beyond the scope of this paper and will be addressed in future studies.

We have also experimented with utilizing a simple threshold-based detection approach to empirically test the advantage of using our graph cuts track segmentation approach. We empirically set the most optimal threshold (running several times and selecting the best threshold) on occupancy likelihoods $\hat{\Theta}$ obtained from the localization algorithm. At each frame, we have regions detected as people. We obtain the connected components on these regions and put a minimum size threshold on these to prune out the noise. Detection error is then computed as specified earlier. Plots of the detection error from the simple thresholding approach are shown in Fig. 10 and can be compared with our approach. As shown by the data, simple thresholding results in many more detection errors compared with our approach, thus corroborating our claim that the integration of detection and tracking in a unified track segmentation formulation is desirable. Though it may be

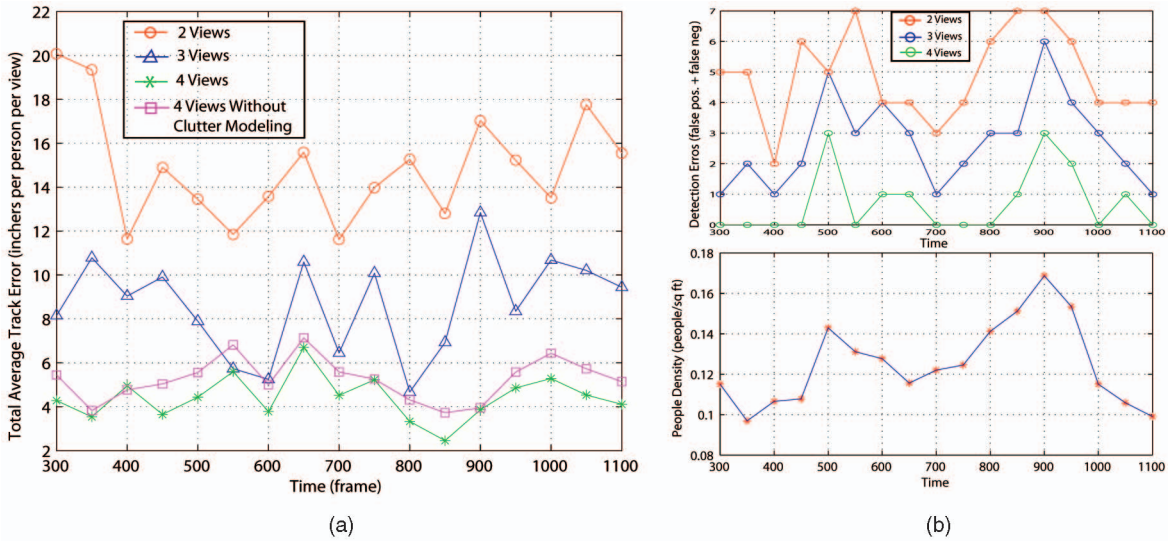

Fig. 9. *Parking lot data set.* (a) Total average track error of persons tracked over time. Pixel distances were converted to inches in the scene using metric calibration data. As expected, track error increases with lesser number of views. This is essentially because of imprecise localization. Also, the track error increases if clutter is not modeled as can be seen for the magenta plot. (b) Plot on the top shows the detection error (number of false positives + number false negatives) over time. The bottom plot shows the variation of the people density over time. Notice the correlation between detection error and people density. As can be seen, increasing density effects the performance of our algorithm. Higher density means more interperson occlusions for any vantage point and, thus, more detection errors.

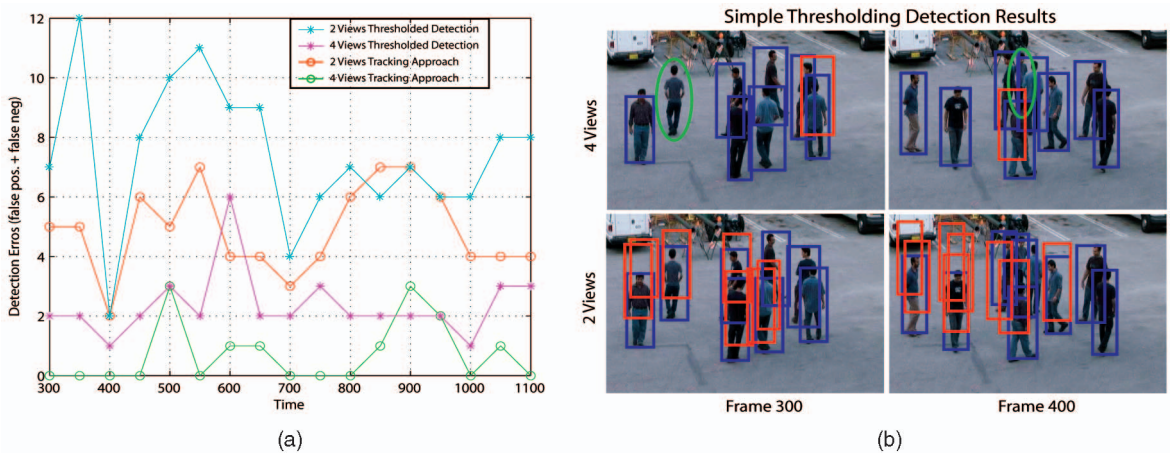

Fig. 10. *Parking lot data set.* (a) Detection error for utilizing a simple thresholding of the occupancy likelihood data compared with our trajectory segmentation-based approach. Plots for detection error using two views and four views are shown. As can be clearly seen, our approach performs much better. (b) Detection results using a threshold-based approach. Blue rectangles are true positives, red rectangles are false positives, and green ellipses are false negatives.

argued that threshold-based detection results can be improved by incorporating more sophisticated models, like human shape priors, we maintain that such models can be used to augment our track segmentation approach (see [14] for a use of shape priors in graph cuts-based segmentation).

Indoor data set. This sequence was captured with four frame synchronized cameras (480 $\times$ 720) placed roughly evenly in a semicircular arc configuration. All four cameras were approximately at head level $\sim$1.8 m. Due to the camera orientation and configuration, the ground plane is only partially visible in just one view. This meant we could not use the ground plane as the reference plane due to a lack of correspondences for homography calculation, a case that might arise in practical scenarios. Therefore, we use the back wall in the scene, which is clearly visible in all the views, as the reference plane. Localization was performed on a total of 20 planes including the wall reference plane and planes parallel to it in the normal direction. There are several straight lines in the direction normal to the back wall plane in the scene (cavities on the right and left of the wall plane, stair case, ceiling, windows, etc.), which are used for the vanishing point calculation. Fig. 11 shows the tracking results for this sequence. The first row shows the top view with the camera configuration and overlaps in the FOVs. As stated for the parking lot sequence, yellower regions have higher camera overlap and the blue squares are tracked locations.

Fig. 12 shows the quantitative analysis for this sequence. We did not have metric calibration data for these sequences; therefore, we calculated the total average track error in the image space. This was done by calculating the distance in pixels between the top of the tracked localization (centroid of top patch of the track bounding cubes in Fig. 11) and the manually marked top of the heads of the people. Fig. 12a shows the variation of the total average track error for a sequence of frames by selectively varying the number of

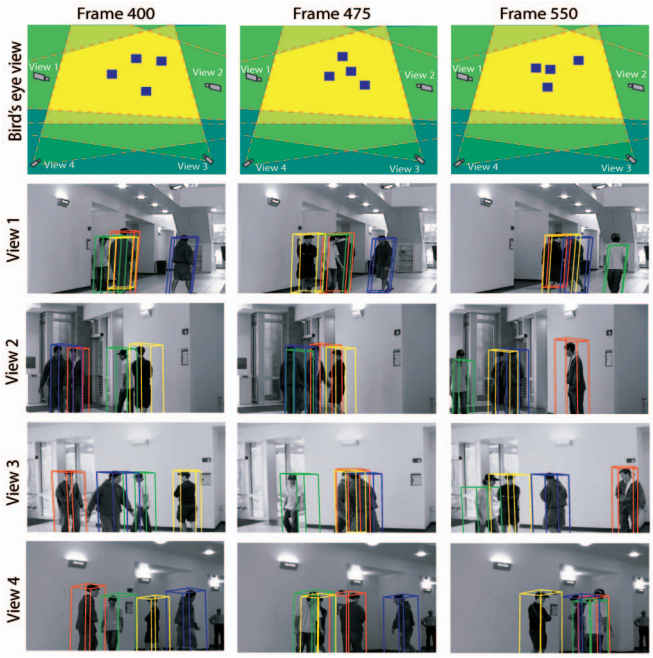

Fig. 11. *Indoor data set*. Tracking using the back wall as the primary reference plane. Twenty planes parallel to the back wall were used in total. The top view color coding is the same as in Fig. 8. Three-dimensional bounding boxes encapsulating the localization on all fusion planes are plotted.

views. With four views, the track error hovers around 20 pixels, which is quite good considering the size of a person is about 250 × 75 pixels and only the head location rather than a central axis was used to calculate the track error. However, with only two views, the tracking becomes intractable. The magenta plot in Fig. 12a also shows the track error for four views if clutter modeling is not used. As shown, total average track error can be reduced by nearly 10 pixels if clutter is modeled. In Fig. 12b, we show the plots of *accumulated* detection errors, the sum of false positives and false negatives accumulated over time, if a single fusion plane is used for the homographic occupancy constraint.

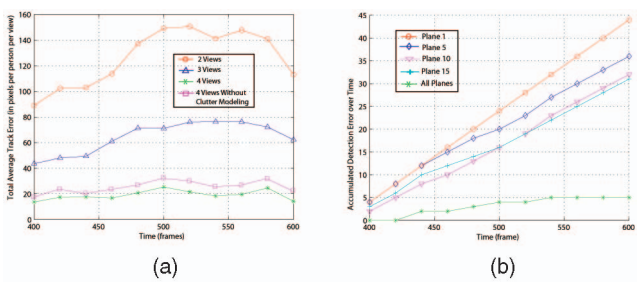

Fig. 12. *Indoor data set analysis*. (a) Total average track error over time from the top center of the track bounding box to the manually marked head locations of people. (b) Plot on the top shows the accumulated detection error (number of false positives + number of false negatives accumulated over time) for different individual planes. Error is the worst for plane 1 (i.e., the back wall) since at no time are people touching (occupying) the back wall. Other planes parallel to the back wall in the normal direction are only marginally better. This is because people keep moving away and toward the back wall in circles, meaning there is no single plane that can be used to reliably localize the people. Since we use all the plane simultaneously, our localization errors are significantly reduced, as shown by the green plot.

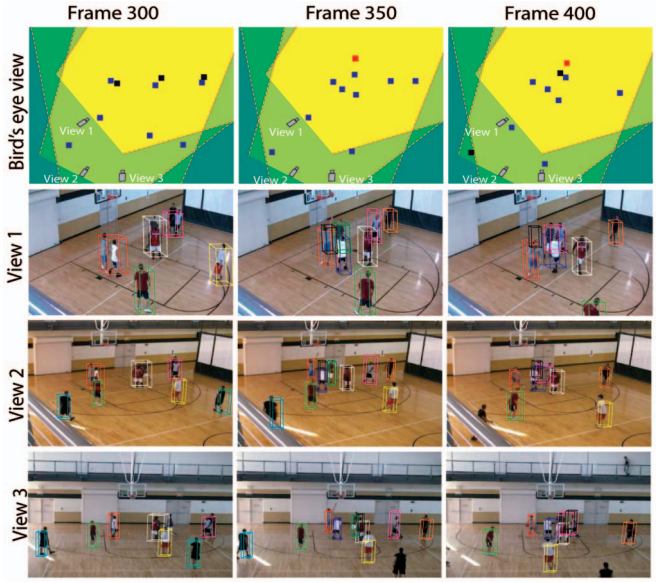

Fig. 13. *Basketball data set*. Tracking of multiple players in a basketball game. Notice the track of the player who is jumping (red track box in frame 350). Due to limitations in the number of cameras and constraints on camera configuration as well as scene clutter (due to reflections off ground and occlusions), our results had relatively higher detection errors. Note the red, white, and magenta track boxes in views corresponding to frame 300. One player is missed in each box (black squares in top view). Also, there are some false positives in frames 350 and 400 (red squares in top view and black track boxes in camera views).

Note that the accumulated detection error contains no extra information from detection error, we use it for better visualization in the plots in this case. Results of 4 of the 20 planes, including the back wall plane and planes parallel to it, are shown. The original reference plane, the back wall, has the worst performance (red plot) because at no time instant were there people touching (occupying) the back wall, resulting in zero detections. Other individual planes fare only marginally better. People kept moving in circles, coming closer and going farther away from the back wall. This meant there was no single plane that could reliably localize all of the people over a meaningful period of time. Fig. 12b also shows in green the plot of accumulated localization errors when using all 20 fusion planes together. As shown, the error is significantly reduced, thus corroborating our initial motivation to use multiple planes to localize people.

Basketball data set. This data set was captured using three cameras (480 × 720) that were arranged roughly in a semicircular arc. The sequences are approximately 1,000 frames long and consists of 10 players playing basketball. Homography constraint fusion was applied at 20 planes including the ground plane and planes parallel to it in the up direction. Based on observations, the highest plane was approximately 2.5 m above the ground.

This data set is challenging not only because of the limited number of views and occlusions due to the high number of players but also because of shadows and reflections off the shiny floor. Moreover, the players have highly nonlinear and unpredictable motion paths including jumps and leaps off the ground. For other approaches, even if the occlusions are resolved and the shadows and reflections removed, it will still be difficult to keep track

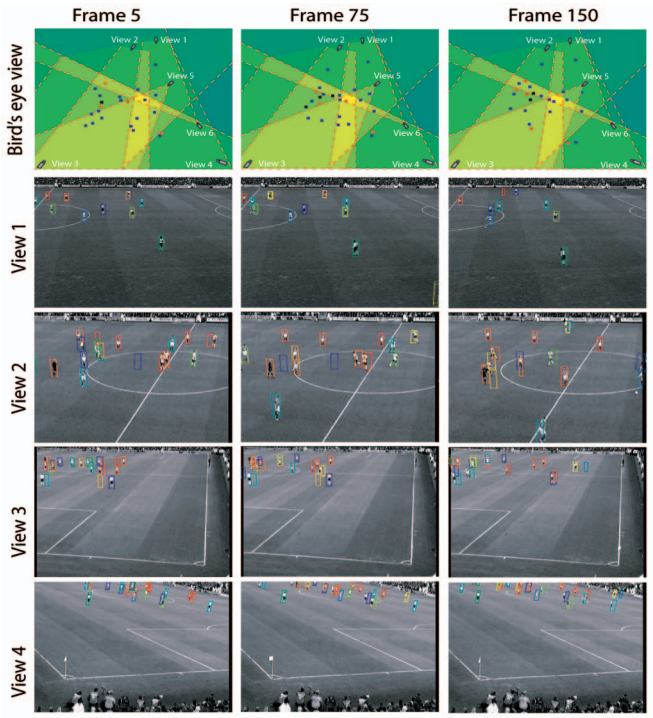

Fig. 14. *Soccer data set*: Tracking of multiple players in a soccer match. The top view is color coded as described in earlier figures. In rows 2-5, we show views 1-4 of the available views. Due to space limitations, all views could not be shown.

of the players with such motion paths, whereas our approach performs quite well (see Fig. 13). Notice the player who is jumping and being tracked (red bounding box, second column of Fig. 13). Clearly, using a single fusion plane, like the ground, would cause the jumping player's track to be lost. Although there are false positives and false negatives (red and black squares in the top view), we believe more views can resolve many of these.

Soccer data set. This a publicly available data set, http://sceptre.king.ac.uk/sceptre/default.html, consisting of eight frame-synchronized views of a soccer match. The cameras are placed in a configuration that covers the entire pitch, with two focused on the goal areas on opposite sides. The sequences are about 1,000 frames long. Occlusions are quite abundant due to the large number of players. There is also a lot of clutter due to jitter of the cameras. We believe this is a result of wind or the shaking of the platform on which the cameras were mounted. Another challenge is the lack of pixel resolution on players. Depending on the view, player patches could be as small as 5×25 pixels. In spite of these challenges, our method was able to localize and track the players with a high degree of accuracy. Fig. 14 shows our tracking results. The first row is the top view, with the same color coding as previously described for other sequences. Notice that there are greater instances of errors in regions of less view overlap and higher density of players.

6.2 Discussion of Failure Scenarios

In addition to the qualitative description and quantitative evaluation, we would also like to summarize the failure scenarios for our approach [60]. If a person is not part of the high foreground likelihood regions in the views, it may cause a missed detection (false negative). This may occur if the person's appearance is very similar to the background or if the person is occluded by some portion of the background itself, e.g., a tree or a part of a building or by another person whose appearance is very similar to the background. Another failure scenario is if a part of the scene is occluded in all views by the foreground objects (other people). In this case, our approach may generate a detection even if the region is not occupied by a person (false positive). Finally, track IDs may be switched in the case when two or more occupancy tracks of different people merge for longer than the sliding window time and then split. This may happen at high people densities when people start touching/bumping into each other or when there no sufficient number of views to "see" the empty spaces between people. Though a simple appearance-based heuristic may be used to resolve the switching of identities (albeit requiring a visibility/occlusion check), we have kept our approach purely occupancy based and this failure case remains.

7 Conclusions

We have presented an algorithm that can reliably track multiple people in a complex environment. This is achieved by resolving occlusions and localizing people on multiple scene planes using a planar homographic occupancy constraint. By combining foreground likelihood information from multiple views and obtaining the global optimum of space-time scene occupancies over a window of frames, we segment out the individual trajectories of the people.

There are many possible extensions of this work. One direction is to incorporate color models in the detection and tracking of individual people. The color models can be used to disambiguate tracks in cases when two or more people come too close to be segmented as separate entities. Using articulated human shape models can be another addition that can act as a prior to prune out false detections and increase robustness of localization. Similarly, a human motion model that takes into account the consistency of speed and direction as well as modeling collision avoidance strategies between people could be an interesting addition. These models may be useful in situations where crowd densities increase and camera views are limited.

Appendix

Let $H_{i_\pi j}$ be the homography between views i and j induced by scene plane π. Now, $H_{i_\pi j}$ can be decomposed as the product of two homographies first from i to π and then from π to j:

$$H_{i_\pi j} = [H_{\pi_{to}j}][H_{i_{to}\pi}]. \tag{11}$$

Similarly, the homography $H_{i_\phi j}$ induced by a plane ϕ that is parallel to π can be written as

$$H_{i_\phi j} = [H_{\phi_{to}j}][H_{i_{to}\phi}]. \tag{12}$$

Now, from Criminisi et al. [8], we have

$$H_{\phi_{to}j} = [H_{\pi_{to}j}] + [0|\gamma \mathbf{v}_{ref}], \tag{13}$$

where $\mathbf{v}_{ref}$ is the vanishing point of the normal direction and γ is a scalar multiple:

$$H_{i_{to}\phi} = inv(H_{\phi_{to}i}) = inv\big([H_{\pi_{to}i}] + [0|\gamma\mathbf{v}_{ref}\big) = H_{i_{to}\pi} - \frac{1}{1+g}[H_{i_{to}\pi}][0|\gamma\mathbf{v}_{ref}][H_{i_{to}\pi}], \quad (14)$$

where $g = trace([0|\gamma\mathbf{v}_{ref}][H_{i_{to}\pi}])$. Replacing (13) and (14) into (12), we have

$$H_{i_\phi j} = \big(H_{\pi_{to}j} + [0|\gamma\mathbf{v}_{ref}]\big)\left(H_{i_{to}\pi} - \frac{1}{1+g}[H_{i_{to}\pi}][0|\gamma\mathbf{v}_{ref}][H_{i_{to}\pi}]\right). \quad (15)$$

Since $H_{i_{to}\pi}$ is a central projection from one plane to another (2D perspectivity with six DOF), the last row is [0 0 1]; therefore, $g = trace([0|\gamma\mathbf{v}_{ref}][H_{i_{to}\pi}]) = \gamma$. Plugging this and (11) into (15) and with some matrix algebra, we reach

$$H_{i_\phi j} = \big(H_{i_\pi j} + [0|\gamma\mathbf{v}_{ref}]\big)\left(I_{3\times3} - \frac{1}{1+\gamma}[0|\gamma\mathbf{v}_{ref}]\right). \quad (16)$$

ACKNOWLEDGMENTS

The authors would like to thank Pavel Babenko for help in the GPU implementation. This research was funded in part by the US Government VACE program.

REFERENCES

[1] A. Azarbayejani and A. Pentland, "Real-Time Self-Calibrating Stereo Person Tracking Using 3D Shape Estimation from Blob Features," *Proc. 13th Int'l Conf. Pattern Recognition,* 1996.

[2] J. Berclaz, F. Fleuret, and P. Fua, "Robust People Tracking with Global Trajectory Optimization," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.

[3] Y. Boukov and M. Jolly, "Interactive Graph Cuts for Optimal Boundary and Region Segmentation of Objects in N-D Images," *Proc. Eighth IEEE Int'l Conf. Computer Vision,* 2001.

[4] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, 2001.

[5] G. Brostow and R. Cipolla, "Unsupervised Bayesian Detection of Independent Motion in Crowds," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.

[6] Q. Cai and J. Aggarwal, "Automatic Tracking of Human Motion in Indoor Scenes Across Multiple Synchronized Video Streams," *Proc. Sixth IEEE Int'l Conf. Computer Vision,* 1998.

[7] T. Chang and S. Gong, "Tracking Multiple People with a Multi-Camera System," *Proc. IEEE Workshop Multi-Object Tracking,* 2001.

[8] A. Criminisi, I. Reid, and A. Zisserman, "Single View Metrology," *Int'l J. Computer Vision,* 1989.

[9] T. Darrell, D. Demirdjian, N. Checka, and P. Felzenszwalb, "Plan-View Trajectory Estimation with Dense Stereo Background Models," *Proc. Eighth IEEE Int'l Conf. Computer Vision,* 2001.

[10] S. Dockstader and A. Tekalp, "Multiple Camera Fusion for Multi-Object Tracking," *Proc. IEEE Workshop Multi-Object Tracking,* 2001.

[11] A. Elfes, "Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation," PhD dissertation, 1989.

[12] J. Franco and E. Boyer, "Fusion of Multi-View Silhouette Cues Using a Space Occupancy Grid," *Proc. 10th IEEE Int'l Conf. Computer Vision,* 2005.

[13] D. Freedman and M.W. Turek, "Illumination-Invariant Tracking via Graph Cuts," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[14] D. Freedman and T. Zhang, "Interactive Graph Cut Based Segmentation with Shape Priors," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[15] J. Gibson, *The Ecological Approach to Visual Perception.* Houghton Mifflen, 1979.

[16] M. Han, W. Xu, H. Tao, and Y. Gong, "An Algorithm for Multiple Object Trajectory Tracking," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.

[17] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real-Time Surveillance of People and Their Activities," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, 2000.

[18] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision.* Cambridge Univ. Press, 2002.

[19] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, and S. Maybank, "Principal Axis-Based Correspondence between Multiple Cameras for People Tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, 2006.

[20] Y. Huang and I. Essa, "Tracking Multiple Objects Through Occlusions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[21] M. Isard and J. MacCormick, "Bramble: A Bayesian Multiple-Blob Tracker," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2001.

[22] R. Jain and K. Wakimoto, "Multiple Perspective Interactive Video," *Proc. IEEE Int'l Conf. Multimedia Computing and Systems,* 1995.

[23] N. Jojic and B. Frey, "Learning Flexible Sprites in Video Layers," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2001.

[24] J. Kang, I. Cohen, and G. Medioni, "Continuous Tracking within and across Camera Streams," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2003.

[25] P. Kelly, A. Katkere, D. Kuramura, S. Moezzi, S. Chatterjee, and R. Jain, "An Architecture for Multiple Perspective Interactive Video," *Proc. Third ACM Int'l Conf. Multimedia,* 1995.

[26] S. Khan and M. Shah, "Consistent Labeling of Tracked Objects in Multiple Cameras with Overlapping Fields of View," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, 2003.

[27] S.M. Khan and M. Shah, "A Multi-View Approach to Tracking People in Crowded Scenes Using a Planar Homography Constraint," *Proc. Ninth European Conf. Computer Vision,* 2006.

[28] Z. Khan, T.R. Balch, and F. Dellaert, "An MCMC-Based Particle Filter for Tracking Multiple Interacting Targets," *Proc. Eighth European Conf. Computer Vision,* 2004.

[29] K. Kim and L. Davis, "Multi-Camera Tracking and Segmentation of Occluded People on Ground Plane Using Search-Guided Particle Filtering," *Proc. Ninth European Conf. Computer Vision,* 2006.

[30] P. Kornprobst and G. Medioni, "Tracking Segmented Objects Using Tensor Voting," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2000.

[31] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, "Multi-Camera Multi-Person Tracking for Easy Living," *Proc. Third IEEE Int'l Workshop Visual Surveillance,* 2000.

[32] D. Lowe, "Distinctive Image Features from Scale Invariant Keypoints," *Int'l J. Computer Vision,* 2004.

[33] F. Lv, T. Zhao, and R. Nevatia, "Self-Calibration of a Camera from Video of a Walking Human," *Proc. 16th Int'l Conf. Pattern Recognition,* 2002.

[34] J. MacCormick and A. Blake, "A Probabilistic Exclusion Principle for Tracking Multiple Objects," *Int'l J. Computer Vision,* 2000.

[35] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* W.H. Freeman, 1982.

[36] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking Groups of People," *Computer Vision and Image Understanding,* 2000.

[37] A. Mittal and S. Larry, "M2tracker: A Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene," *Int'l J. Computer Vision,* 2002.

[38] A. Nakazawa, H. Kato, and S. Inokuchi, "Human Tracking Using Distributed Vision Systems," *Proc. 14th Int'l Conf. Pattern Recognition,* 1998.

[39] U. Neisser, *Cognition and Reality: Principles and Implications of Cognitive Psychology.* W.H. Freeman, 1976.

[40] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe, "A Boosted Particle Filter: Multitarget Detection and Tracking," *Proc. Eighth European Conf. Computer Vision,* 2004.

[41] J. Orwell, S. Massey, P. Remagnino, D. Greenhill, and G. Jones, "A Multi-Agent Framework for Visual Surveillance," *Proc. IEEE Int'l Conf. Image Processing,* 1999.

[42] S. Park and M.M. Trivedi, "Analysis and Query of Person-Vehicle Interactions in Homography Domain," *Proc. Fourth ACM Int'l Workshop Video Surveillance and Sensor Networks,* 2006.

[43] S. Park and M.M. Trivedi, "Multi-Perspective Video Analysis of Persons and Vehicles for Enhanced Situational Awareness," *Proc. IEEE Int'l Conf. Intelligence and Security Informatics,* 2006.
[44] A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, and W. Hu, "Multi-Object Tracking Through Simultaneous Long Occlusions and Split and Merge Conditions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.
[45] A. Poore, "Multidimensional Assignments and Multitarget Tracking," *Proc. DIMACS Workshop,* 1995.
[46] D. Reid, "An Algorithm for Tracking Multiple Targets," *IEEE Trans. Automatic Control,* 1979.
[47] R. Rosales and S. Sclaroff, "3D Trajectory Recovery for Tracking Multiple Objects and Trajectory Guided Recognition of Actions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 1999.
[48] C. Rother, "A New Approach for Vanishing Point Detection in Architectural Environments," *Proc. 13th British Machine Vision Conf.,* 2002.
[49] S. Rotman, G. Tidhar, and M. Kowalczyk, "Clutter Metrics for Target Detection Systems," *IEEE Trans. Aerospace and Electronic Systems,* 1994.
[50] K. Sato, T. Maeda, H. Kato, and S. Inokuchi, "Cad-Based Object Tracking with Distributed Monocular Camera for Security Monitoring," *Proc. Second IEEE CAD-Based Vision Workshop,* 1994.
[51] D. Schmieder and M. Weathersby, "Performance Detection with Variable Clutter Resolution," *IEEE Trans. Aerospace and Electronic Systems,* 1983.
[52] A. Senior, A. Hampapur, Y. Tian, L. Brown, S. Pankanti, and R. Bolle, "Appearance Models for Occlusion Handling," *Proc. Second IEEE Workshop Performance Evaluation of Tracking and Surveillance,* 2001.
[53] I. Sethi and R. Jain, "Finding Trajectories of Feature Points in a Monocular Image Sequence," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 9, 1987.
[54] H. Sidenbladh, M. Black, and D. Fleet, "Stochastic Tracking of 3D Human Figures Using 2D Image Motion," *Proc. Sixth European Conf. Computer Vision (ECCV),* 2000.
[55] C. Stauffer and W. Grimson, "Adaptive Background Mixture Models for Real-Time Tracking," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 1999.
[56] H. Tao, H.S. Sawhney, and R. Kumar, "Object Tracking with Bayesian Estimation of Dynamic Layer Representations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, 2002.
[57] S. Thrun, "Learning Occupancy Grid Maps with Forward Sensor Models," *J. Autonomous Robots,* 2003.
[58] C. Wren, A. Azarbayejani, T. Darell, and A. Pentland, "Pfinder: Real-Time Tracking of Human Body," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, 1997.
[59] Y. Wu, T. Yu, and G. Hua, "Tracking Appearances with Occlusions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2003.
[60] D.B. Yang, H.H. Gonzalez-Banos, and L.J. Guibas, "Counting People in Crowds with a Real-Time Network of Simple Image Sensors," *Proc. Ninth IEEE Int'l Conf. Computer Vision,* 2003.
[61] A. Yilmaz, O. Javed, and M. Shah, "Object Tracking: A Survey," *ACM J. Computing Surveys,* 2006.
[62] A. Yilmaz, X. Li, and M. Shah, "Contour-Based Object Tracking with Occlusion Handling in Video Acquired Using Mobile Cameras," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, 2004.
[63] T. Zhao and R. Nevatia, "Tracking Multiple Humans in Complex Situations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, 2004.
[64] T. Zhao and R. Nevatia, "Tracking Multiple Humans in Crowded Environment," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.

Saad M. Khan received the BS degree in computer system engineering from the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan, in 2003. He received the PhD degree from the Computer Vision Laboratory at the University of Central Florida in 2008. His research interests include visual tracking, 3D reconstruction, activity recognition, and vision-based navigation of UAVs. He is currently a member of the technical staff at Sarnoff Corp.

Mubarak Shah is the Agere Chair Professor of Computer Science and the founding director of the Computer Visions Lab at the University of Central Florida (UCF). He has coauthored two books: *Motion-Based Recognition* (1997) and *Video Registration* (2003), both by Kluwer Academic. In 2006, he was awarded a Pegasus Professor award, the highest award at UCF, given to a faculty member who has made a significant impact on the university, has made an extraordinary contribution to the university community, and has demonstrated excellence in teaching, research, and service. He was an IEEE Distinguished Visitor speaker for 1997-2000 and received the IEEE Outstanding Engineering Educator Award in 1997. He received Harris Corp.'s Engineering Achievement Award in 1999, the TOKTEN awards from UNDP in 1995, 1997, and 2000, the Teaching Incentive Program award in 1995 and 2003, the Research Incentive Award in 2003, the Millionaires' Club awards in 2005 and 2006, the University Distinguished Researcher award in 2007, an honorable mention for the 10th IEEE International Conference on Computer Vision's Where Am I? Challenge Problem, and was nominated for the best paper award for the ACM Multimedia Conference in 2005. He is an editor of an international book series on video computing, the editor-in-chief of *Machine Vision and Applications* journal, and an associate editor of *ACM Computing Surveys* journal. He was an associate editor of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* and a guest editor of the special issue of the *International Journal of Computer Vision* on video computing. He is a fellow of the IEEE, the International Association for Pattern Recognition (IAPR), and the Society of Photo-Optical Instrumentation Engineers (SPIE).