# Homography Based Multiple Camera Detection and Tracking of People in a Dense Crowd

Ran Eshel and Yael Moses
Efi Arazi School of Computer Science,
The Interdisciplinary Center, Herzliya 46150, Israel

## Abstract

*Tracking people in a dense crowd is a challenging problem for a single camera tracker due to occlusions and extensive motion that make human segmentation difficult. In this paper we suggest a method for simultaneously tracking all the people in a densely crowded scene using a set of cameras with overlapping fields of view. To overcome occlusions, the cameras are placed at a high elevation and only people's heads are tracked. Head detection is still difficult since each foreground region may consist of multiple subjects. By combining data from several views, height information is extracted and used for head segmentation. The head tops, which are regarded as 2D patches at various heights, are detected by applying intensity correlation to aligned frames from the different cameras. The detected head tops are then tracked using common assumptions on motion direction and velocity. The method was tested on sequences in indoor and outdoor environments under challenging illumination conditions. It was successful in tracking up to 21 people walking in a small area (2.5 people per $m^2$), in spite of severe and persistent occlusions.*

## 1. Introduction

People tracking is a well-studied problem in computer vision, mainly, but not exclusively, for surveillance applications. In this paper we present a new method for tracking multiple people in a dense crowd by combining information from a set of cameras overlooking the same scene. The main challenge encountered by tracking methods is the severe and persistent occlusion prevalent in images of a dense crowd (as shown in Fig. 1). Most existing tracking methods use a single camera, and thus do not cope well with crowded scenes. For example, trackers based on a human shape model such as Rodriguez & Shah [18] or Zhao & Nevatia [23] will encounter difficulties since body parts are not isolated, and may be significantly occluded. Multiple camera tracking methods often perform segmentation in each view separately, and are thus susceptible to the same problems (e.g., [11, 15]).

Our method avoids occlusion by only tracking heads. We place a set of cameras at a high elevation, from which the heads are almost always visible. Even under these conditions, head segmentation using a single image is challenging, since in a dense crowd, people are often merged into large foreground blobs (see Fig. 4). To overcome this problem, our method combines information from a set of static, synchronized and partially calibrated cameras, with overlapping fields of view (see examples in Fig. 1).

We rely on the assumption that the head is the highest region of the body. A head top forms a 2D blob on the plane parallel to the floor at the person's height. The set of frames taken from different views at the same time step is used to detect such blobs. For each height, the foreground images from all views (each may be a blob containing many people) are transformed using a planar homography [3] to align the projection of the plane at that height. Intensity correlation in the set of transformed frames is used to detect the candidate blobs. In Fig. 2 we demonstrate this process on a scene with a single person. Repeating this correlation for a set of heights produces 2D blobs at various heights that are candidate head tops. By projecting these blobs to the floor, multiple detections of the same person at different heights can be removed. At the end of this phase we obtain, for each time step, the centers of the candidate head tops projected to the floor of a reference sequence.

In the next phase of our algorithm, the detected head top centers are combined into tracks. At the first level of tracking, atomic tracks are detected using conservative assumptions on the expected trajectory, such as consistency of motion direction and velocity. At the second level, atomic tracks are combined into longer tracks using a score which reflects the likelihood that the two tracks belong to the same trajectory. Finally, a score function based on the length of the trajectory and on the consistency of its motion is used to detect false positive tracks and filter them out.

Our method overcomes hard challenges of tracking people: severe and persistent occlusions, subjects with non-

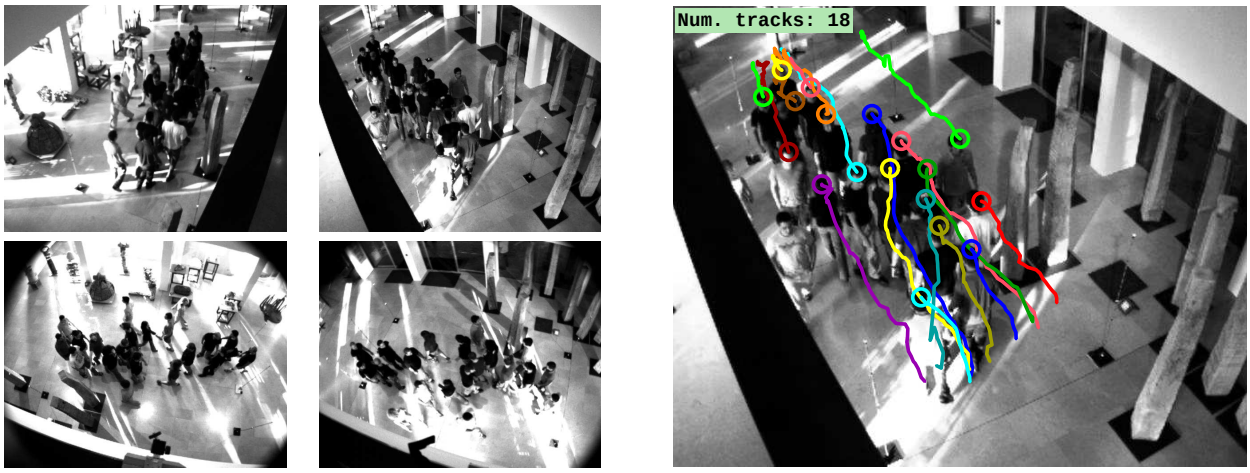



Figure 1. Four views of the same scene, with tracking result on the reference frame.

standard body shape (e.g., a person carrying a suitcase or a backpack), people wearing similar clothing, shadows and reflections on the floor, highly varied illumination within the scene, and poor image contrast. The method was tested on indoor and outdoor sequences with challenging lighting conditions, and was successful in tracking up to 21 people walking in a small area (2.5 people per $m^2$).

## 2. Related Work

Until recent years, the bulk of research in the field of people detection and tracking concentrated on using a single camera to track a small number of subjects. Papageorgiou et al. [16] use SVM detectors based on Haar wavelets. Felzenszwalb [4] trains a classifier using a human shape model. Both methods are trained on full human figures, and will not perform well if subjects are even partially occluded. Leibe et al. [14] also use a full-body representation, but increase its flexibility by allowing interpolation between local parts seen on different training objects. Wu & Nevatia [21] detect body parts by boosting a number of weak classifiers, and track partially occluded humans using data association and mean shift. Viola et al. [20] combine motion and appearance for segmenting pedestrians. Several methods employ a Bayesian framework, using Kalman filters or particle filters for tracking: Isard & MacCormick [8] handle occlusions using a 3D object model that provides depth ordering; Zhao et al. [23] use a coarse 3D human shape model to separate between different people that belong to a single foreground blob; Smith et al. [19] and Yu et al. [22] use sophisticated background subtraction methods for detection, and an MCMC approach to sample the solution space efficiently.

These and other single camera methods are inadequate for handling highly dense crowds such as those considered in this paper, due to severe occlusion which results in large foreground blobs comprised of multiple people. For example, a suggested comparison between our method and the state-of-the-art single view tracking system developed by Wu, Zhao & Nevatia could not be performed, since their method was reported to be inapplicable under these challenging density and illumination conditions. [1]

Multiple cameras were traditionally used in tracking for extending the limited viewing area of a single camera. In this case, tracking is performed separately for each camera, and the responsibility of tracking a given subject is transfered from one camera to another [1, 17]. Some methods use multiple cameras with overlapping fields of view. Krumm et al. [11] use pairs of cameras to resolve ambiguity using 3D stereo information. Their method is based on background subtraction, and hence is limited when a dense crowd is considered. Mittal & Davis [15] employ a higher level of collaboration between cameras, where foreground blob ambiguity is resolved by matching regions along epipolar lines. The main limitation of this method is its reliance on the assumption that different people within a single foreground blob are separable based on color segmentation alone. This assumption does not always hold, since people often wear similarly colored clothes. Fleuret et al. [5] combine a generative model with dynamic programming, and demonstrate tracking of up to six people.

The method most similar to ours for detecting people from multiple cameras was proposed by Khan & Shah [9]. They use a homography transformation to align the foreground of the floor plane from images taken from a set of cameras with overlapping fields of view, and achieve

[1] Personal communication.

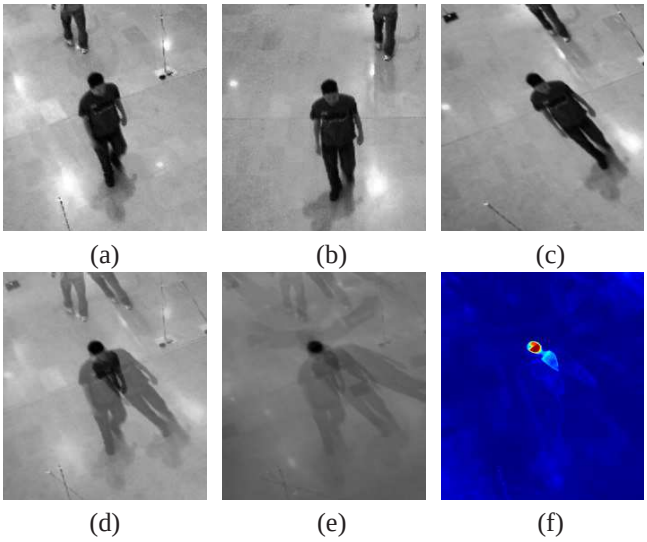


Figure 2. 2D patch detection demonstrated, for clarity, on a single, isolated person. (a,b) Two views of the same person. (c) Homography transformation is applied to image *b* to align points on the 3D plane at the head-top height with their counterparts in image *a*. (d) Image *c* overlaid on image *a*. (e) Overlay of additional transformed images. (f) Variance map of the hyper-pixels of image *e*, color coded such that red corresponds to a low variance.

good results in moderately crowded scenes. However, their method seems inadequate for handling highly crowded scenes. On one hand, tracking people's feet rather than their heads precludes the use of intensity value correlation, since the occlusion of the feet in a dense crowd is likely to cause many false negative detections. On the other hand, detection based solely on foreground/background separation of images rather than on a more discriminative correlation of intensity values can result in false positive detections (as explained in Sec. 3.1.3, and demonstrated in Fig. 4**b**).

Recently, Khan et. al. [10] suggested applying the same concept to planes at multiple heights for 3D shape recovery of non-occluded objects. Several other methods have utilized multiple cameras viewing a single object from different directions for 3D reconstruction, based on the visual hull concept (Laurentini [12]), or on constructing a space occupancy grid (Cheung et al. [2], Franco et al. [6]). However, none of these methods was used for tracking, or in the presence of occlusion.

## 3. The Method

We assume a set of synchronized and partially calibrated cameras overlooking a single scene, where head tops are visible. The partial calibration of the cameras consists of the homography of 3 planes parallel to the floor between each pair of cameras.

Initially, head top centers and their heights are detected (each represented by a single feature point), and projected to the floor. These feature points are then tracked to recover the trajectories of people's motion, and filtered to remove false positives.

### 3.1. Head Top Detection

The head top is defined as the highest 2D patch of a person. The detection of candidate head tops is based on *co-temporal* frames, that is, frames taken from different sequences at the same time. Since we assume synchronized sequences, co-temporal frames are well defined. Fig. 4 shows intermediate results of the method described below.

#### 3.1.1 2D Patch Detection

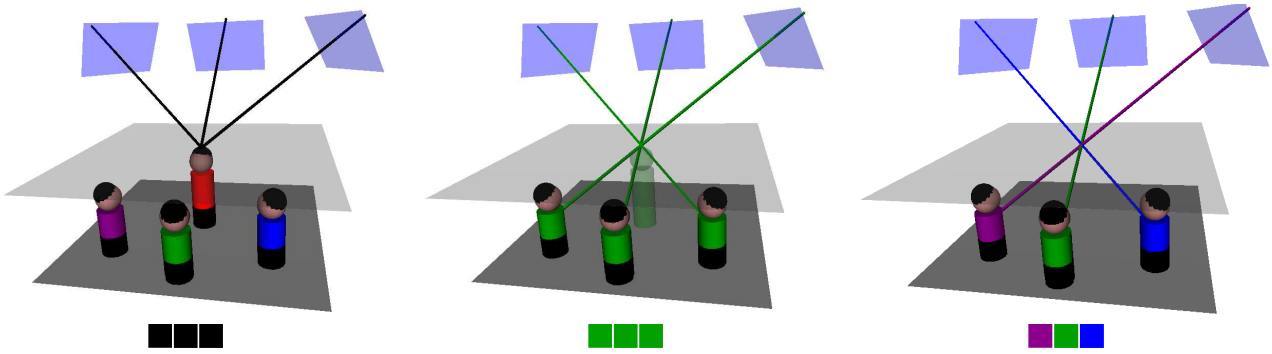
To detect a 2D patch visible in a set of co-temporal frames, we use the known observation that images of a planar surface are related by a homography transformation. When a homography transformation is applied to images of an arbitrary 3D scene, the points that correspond to the plane will align, while the rest of the points will not. This idea is demonstrated in Fig. 2 for a single person at a given height.

Consider $n$ synchronized cameras. Let $S_i$ be the sequence taken by camera $i$, with $S_1$ serving as the reference sequence. Let $\pi^h$ be a plane in the 3D scene parallel to the image floor at height $h$. A *$\pi$-mapping* between an image and a reference image is defined as the homography that aligns the projection of points on the plane $\pi$ in the two images. For a plane $\pi^h$ and sequences $S_i$ and $S_1$, it is given by the $3 \times 3$ homography matrix $A^h_{i,1}$. Using the three known homography matrices given by the partial calibration, $A^{h1}_{i,1}$, $A^{h2}_{i,1}$ and $A^{h3}_{i,1}$, he homography matrices $A^h_{i,1}$ can be computed for any height $h$.

Consider $S_1(t)$, a frame of the reference sequence in time $t$. To detect the set of pixels in $S_1(t)$ that are projections of a 2D patch at height $h$, the co-temporal set of $n$ frames is used. Each of the frames is aligned to the sequence $S_1$, using the homography given by the matrix $A^h_{i,1}$. Let $S_i(t)$ be a frame from sequence $i$ taken at time $t$. Let $p \in S_i(t)$, and let $I_i(p)$ be its intensity. A *hyper-pixel* is defined as an $n \times 1$ vector $\bar{q}^h$ consisting of the set of intensities that are $\pi^h$-mapped to $q \in S_1(t)$. The $\pi^h$-mapping of the point $p \in S_i(t)$ to a point $q$ in frame $S_1(t)$ is given by $q = A^h_{i,1} p_i$. The inverse transformation, $p_i = A^h_{1,i} q$, allows us to compute $\bar{q}^h$:

$$\bar{q}^h = \begin{pmatrix} I_1(q) \\ I_2(p_2) \\ . \\ . \\ I_n(p_n) \end{pmatrix} = \begin{pmatrix} I_1(q) \\ I_2(A^h_{1,2} q) \\ . \\ . \\ I_n(A^h_{1,n} q) \end{pmatrix}.$$

The hyper-pixel $\bar{q}^h$ is computed for each pixel $q \in S_1(t)$. Highly correlated intensities within a hyper-pixel indicate



(a) Low variance (b) Low variance (c) High variance

Figure 3. After applying the plane transformation which corresponds to the imaginary plane in the scene, the hyper-pixel of the aligned images will contain the marked rays. (a) A 3D point at the plane height is detected where a person is present. (b) A false positive detection occurs due to accidental projections of points from different people. This will only happen if all points coincidentally have the same color. (c) In the more common case, points belonging to different objects have different colors. This results in high hyper-pixel intensity variance, which prevents false positive detection.

that the pixel is a projection of a point on the considered plane $\pi^h$. A low correlation can be expected for other points provided that the scene is not homogeneous in color. Using hyper-pixel intensity variance, we obtain a set of pixels that are likely to be projections of points on the plane $\pi^h$. Simple clustering, using double threshold hysteresis on these pixels and a rough estimation of the head top size (in pixels), can be used for detecting candidate 2D patches on the plane $\pi^h$. If a blob is larger than the expected size of a head top, a situation that may occur in extremely dense crowds, the blob is split into several appropriately sized blobs using K-means clustering. The centers of the 2D patches are then used for further processing.

A possible source of false positive detections is homogeneous background. For example, in an outdoor scene, the texture or color of the ground may be uniform, as may be the floor or walls in an indoor scene. We therefore align only the foreground regions, computed using a simple background subtraction algorithm (which subtracts each frame from a single background frame, taken when the scene was empty).

#### 3.1.2 Detecting the Highest 2D Patch

The process of detecting 2D patches is repeated for a set $H = \{h_1, ..., h_n\}$ of expected people heights. The set is taken at a resolution of 5cm. We assume that the head tops are visible to all cameras. It follows that at this stage of our algorithm, all head tops are detected as 2D patches at one of the considered heights. However, a single person might be detected as patches at several heights, and all but the highest one should be removed. To do so, we compute the foot location of each of the 2D patches as would appear in the reference sequence.

The foot location is assumed to be the orthogonal projection of a 2D patch at a given height $h$ to the floor. The projection is computed using a homography transformation from the reference sequence to itself. The homography aligns the location of each point on the plane $\pi^h$ in the reference image with the location of its projection to the plane $\pi^0$ in the same image. For each height $h_i \in H$, the homography transformation that maps the projection of the plane $\pi^{h_i}$ to the floor of sequence $S_1$ is given by the $3 \times 3$ homography matrix $B^{h_i}$. These matrices can be computed on the basis of the partial calibration assumption of our system. For a head top center $q \in S_1(t)$, detected at height $h$, the projection to the floor of $S_1$ is given by $B^{h_i}\, q$. For each floor location, a single 2D patch is chosen. If more than one patch is projected to roughly the same foot location, the highest one is chosen, and the rest are ignored. This provides, in addition to detection, an estimation of the detected person's height, which can later assist in tracking.

#### 3.1.3 Expected 'Phantoms'

Phantoms typically occur when people are dressed in similar colors, and the crowd is dense. As a result, portions of the scene may be homogeneous, and accidental intensity correlation of aligned frames may be detected as head tops. Fig. 3**b** illustrates how plane alignment can correlate non-corresponding pixels originating from different people who happen to be wearing similarly colored clothes. In this case, rays intersect in front of the people, and the created phantom is taller. Similarly, shorter phantoms may appear if the rays intersect behind the people. Note that if only background/foreground values are used, as in [9], such accidental detections will occur even if people are wearing different colors (as in Fig. 3**c**). Our method will not detect a phantom

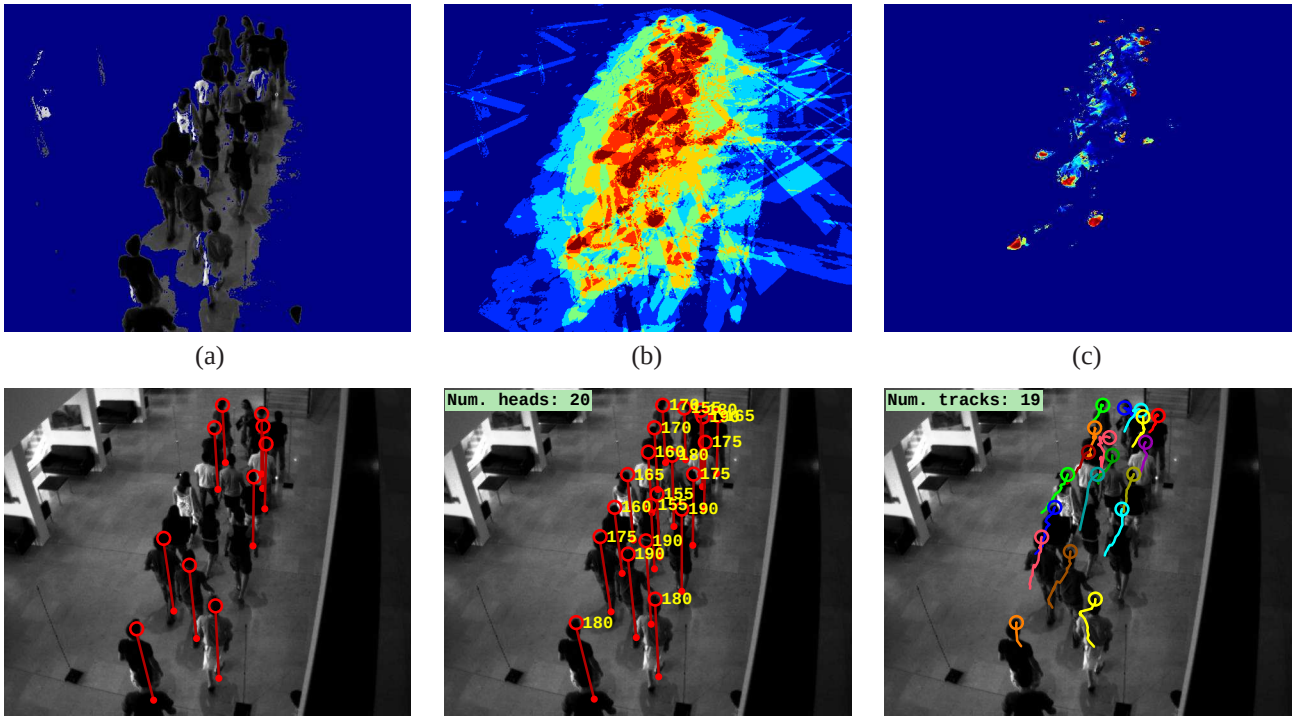



Figure 4. Intermediate results of head top detection. (a) Background subtraction on a single frame. (b) Aligned foreground of all views for a given height (color coded for the number of foregrounds in each hyper-pixel, where red is high). (c) Variance of the foreground hyper-pixels (red for low). (d) Detected head tops at a given height, and their projection to the floor. (e) The same as *d* for all heights. (f) Tracking results with 20 frame history.

in this case, since it uses intensity value correlation.

Phantoms can also affect the detection of real people walking in the scene: the head of a phantom can be just above a real head, causing it to be removed since it is not the highest patch above the foot location. The probability of detecting phantoms can be reduced by increasing the number of cameras (see Sec. 4.3). We remove phantoms in the tracking phase, by filtering out tracks that exhibit abnormal motion behavior. Phantom removal can be further improved by utilizing human shape detection methods, but this is beyond the scope of this paper.

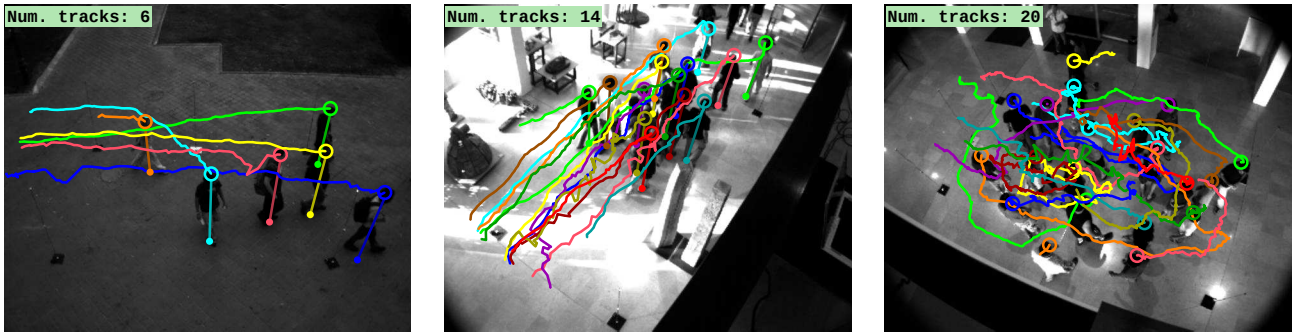
### 3.2. Tracking

The input to the tracker for each time step consists of two lists of head top centers projected to the floor of the reference sequence. Each list is computed using a different threshold. The high threshold list will have less false positive head top detections but more false negative detections than the lower threshold list.

At the first stage of tracking, atomic tracks are computed using prediction of the feature location in the next frame based on its motion velocity and direction in previous ones. Tracking is performed using the high threshold list. If several features are found within a small radius of the expected region, the nearest neighbor is chosen. If no feature is found within the region, the search is repeated using the lower threshold list. Failure to find the feature in either list is considered a negative detection. The termination of tracks is determined by the number of successive negative detections. After all tracks have been matched to features in a given time step, the remaining unmatched features are considered as candidates for new tracks. Tracks are initialized from these candidates only after two or more consecutive positive detections.

The result of the first stage of tracking is a large number of tracks, some of which are fragments of real trajectories and others which are false positives. The next stage combines fragments into long continuous tracks, leaving short unmatched tracks for deletion in the final stage.

Let $tr_i$ and $tr_j$ be two atomic tracks. The numbers of the first and last frames of a track are denoted by $f(tr_i)$ and $\ell(tr_i)$, respectively. The time overlap of two tracks is defined as $overlap(tr_i, tr_j) = f(tr_j) - \ell(tr_i)$. Two tracks, $tr_i$ and $tr_j$, are considered for merging if $-10 \leq overlap(tr_i, tr_j) \leq 40$. A merge score is computed for each pair of tracks that satisfies this condition. The




**S1** **S3a** **S5**

Figure 5. Examples of tracked trajectories from three sequences. (For sequences **S1** and **S3a**, sticks connecting heads and their projections to the floor are displayed. For sequence **S5**, due to the complexity of the trajectories, only heads are displayed.)

score is a function of the following measures: $m_1$ – the amount of overlap between the tracks; $m_2$ – the difference between the two tracks' motion directions; $m_3$ – the direction change required by $tr_i$ in order to reach the merge point with $tr_j$; $m_4$ – the height difference between $tr_i$ and $tr_j$; $m_5, m_6$ – the minimal and average distances between corresponding points along the overlapping segments (or along the expected paths of the trajectories, in case of a negative overlap). The merge score is defined by: $score(tr_i, tr_j) = \frac{1}{6}\sum m_i/\widehat{m}_i$, where $\widehat{m}_i$ is the maximal expected value of the measure $m_i$.

Finally, a consistency score is used to remove tracks that are suspected as false positives. This score is based on weighted components which include the average change in speed, direction and height between any two consecutive time steps, and the track length. This heuristic successfully removes most of the phantom tracks. In addition, pairs of tracks that consistently move together, staying within a very small distance from each other, are detected. In such cases, the shorter track, which is usually the shoulder, is deleted.

## 4. Experimental Results

To demonstrate the effectiveness of our method, we performed experiments on real video sequences under changing conditions. In Sec. 4.2 we describe the scenarios and the results of applying our method to several indoor and outdoor sequences with varying degrees of crowd density and challenging illumination conditions. In Sec. 4.3 we investigate how changing the number of cameras affects the tracking results.

### 4.1. Implementation and System Details

We used between 3 and 9 USB cameras (IDS uEye UI-1545LE-C), connected to 3 laptops. The cameras were placed around the scene, 2-3 meters apart, with the vertical viewing angle of each camera rotated at $30^o$ relative to its neighbor. Horizontally, they were placed at an elevation of 6 meters, viewing the scene at a relatively sharp angle ($45^o$ or more below the horizon). Detection and tracking were performed on an area of $3 \times 6$ meters. All test sequences were taken at a rate of 15 frames per second, with an image size of $640 \times 512$.

The cameras were calibrated using vertical poles placed at the corners of the scene, mounted with LEDS blinking at unique frequencies, as described in [7]. In future work we intend to develop a calibration method that relies on tracked people in a non-dense environment, similar to [13].

The algorithm was implemented in Matlab on gray level images. The algorithm's behavior is controlled by several parameters, all of which have a single global setting except for the hysteresis double thresholds. These are used to isolate high correlation (low variance) hyper-pixels of plane-aligned images, and are set manually for each sequence, since they depend on volatile factors such as the lighting conditions and the number of cameras.

### 4.2. Sequences and Results

Below we describe the different scenarios used for testing our approach, and assess the system's performance.

The following evaluation criteria reflect both the success of recovering each of the trajectories and the success of assigning a single ID to each one. True Positive (*TP*): 75%-100% of the trajectory is tracked, possibly with some ID changes; Perfect True Positive (*PTP*): 100% of the trajectory is tracked, with a single ID (note that these trajectories are counted in TP as well); Detection Rate (*DR*): percent of frames tracked compared to ground truth trajectory, independent of ID change (and including false negatives); ID Changes (*IDC*): number of times a track changes its ID; False Negative (*FN*): less than 75% of the trajectory is tracked; False Positive (*FP*): a track with no real trajectory.

Table 1 summarizes the tracking results. Examples can be seen in Fig. 1 and in Fig. 5, where each detected person

| Seq | GT | TP | PTP | IDC | DR % | FN | FP |
|---|---|---|---|---|---|---|---|
| S1 | 27 | 26 | 23 | 3 | 98.7 | 1 | 6 |
| S2 | 42 | 41 | 39 | 0 | 97.9 | 1 | 5 |
| S3a | 19 | 19 | 19 | 0 | 100.0 | 0 | 0 |
| S3b | 18 | 18 | 18 | 0 | 100.0 | 0 | 2 |
| S3c | 21 | 21 | 20 | 1 | 99.1 | 0 | 0 |
| S4 | 23 | 23 | 22 | 0 | 99.1 | 0 | 1 |
| S5 | 24 | 23 | 14 | 12 | 94.4 | 1 | 0 |
| Total | 174 | 171 | 155 | 16 | 98.4 | 3 | 14 |

Table 1. Tracking results on 7 Sequences (GT – Ground Truth; TP – True Positive, 75%-100% tracked; PTP – Perfect True Positive, 100% tracked, no ID changes along the trajectory; IDC – ID Changes; DR – Detection Rate; FN – False Negative; FP – False Positive).

is marked by his head center. The tails mark the detected trajectories up to the displayed frame.

We next describe each sequence in detail:[2]

**S1:** A long (1500 frames), relatively sparse (up to 6 concurrent trajectories), outdoor sequence using only 6 cameras which, due to physical limitations, are all collinear. The sequence was taken at twilight, and thus suffers from dim lighting and poor contrast. The tracking results are very good, except for a high false positive rate resulting from the low threshold chosen to cope with the low image contrast. Fig. 5**a** presents the tracking results on this sequence.

**S2:** A long (1100 frames) indoor sequence, with medium crowd density using 9 cameras. People in the scene move in groups (up to 9 people concurrently). Lighting conditions are very hard: bright lights coming in through the windows and reflected by the shiny floor create a highly contrasted background; long dark shadows interfere with foreground/background separation; inconsistent lighting within the scene significantly alters an object's appearance along different parts of its trajectory. In addition, tall statues are placed along the path, sometimes causing almost full occlusion. Despite these problems, the tracking quality is good, with only a single track lost, and most of the others perfectly tracked.

**S3:** Three excerpts from a longer sequence (200, 250 and 300 frames) with a very high crowd density, taken with 9 cameras. The scene is the same brightly lighted indoor scenario described in the previous sequence. The sequences contain 57 trajectories in total, with up to 19 concurrent. All of the people move very closely together in a single group and in the same direction (**S3a** & **S3b**), or split into two groups which pass close to each other in opposite directions (**S3c**). An additional difficulty is the inclusion of several bald-headed people in the sequence: the bright overhead lights falling on a bald head give it a different appearance in different views, resulting in a high hyper-pixel variance and a detection failure. Despite similar density, tracking results are significantly better than in sequence **S5**, partly because of the higher number of cameras, but mostly because of the more natural motion patterns displayed by the people. The detection rate is almost perfect (99.7%), and the error rate is very low (a total of 2 false positives, 0 false negatives and 2 ID changes for the three sequences combined). Fig. 5**b** presents the tracking results on sequence **S3a**.

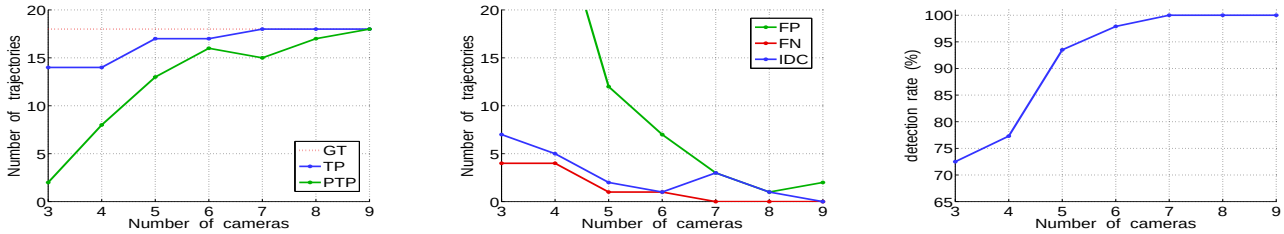
**S4:** A high crowd density sequence (200 frames), taken using 6 cameras placed around the scene. Most of the people are visible at the same time (up to 19), and all of them move in the same direction, making separation based on motion impossible. Tracking results are very good: one of the tracks is detected late (30 frames after first appearing), while all the others are perfectly tracked.

**S5:** A very high crowd density sequence (200 frames) with complex motion taken with the same setup as above. The sequence begins with 21 people crowded into an $8m^2$ area, a density of over 2.5 people per $m^2$. People then start to move in an unnaturally complex manner – changing directions sharply and frequently, and passing very close to each other. The detection results are good, with a 94.4% detection rate and no false positives, but the tracking consistency is not as good, with almost half of the trajectories changing their ID at some point along their path. Fig. 5**c** presents the tracking results on this sequence. The tails demonstrate the complex motion of the people.

### 4.3. Varying the Number of Cameras

In theory, two or three cameras are sufficient for applying our method. In this experiment we test the effect of varying the number of cameras in one of our more challenging sequences, **S3b**. The results are summarized in Fig. 6. In general, both detection and tracking quality improve as the number of cameras increases. However, increasing this number beyond six has a negligible effect. The detection rate and the true positive detection remain high even when the number of cameras is decreased to three. As mentioned in Sec. 3 and demonstrated in Fig. 3**b**, decreasing the number of cameras may increase the number of accidental matchings, causing phantoms to appear. The effect of this phenomenon is apparent in Fig. 6**b**. The ambiguity caused by the presence of a large number of phantoms also affects other parameters, resulting in an increase in the number of ID changes and of false negative detections. We can therefore conclude that our tracker performs well when the number of cameras is sufficient for handling the crowd density. Otherwise, its performance gradually degrades as the number of cameras decreases.

[2]Tracking results can be seen in:
ftp://ftp.idc.ac.il/Pub/Users/CS/Yael/CVPR-2008/CVPR-2008-results.zip




(a) Ground truth and true positives (b) False positives, false negatives and ID changes (c) Detection rate

Figure 6. System performance as a function of the number of cameras. Results improve as the number of cameras increases. When this number drops below 5, system performance deteriorates considerably.

## 5. Conclusions

We suggest a method based on a multiple camera system for tracking people in a dense crowd. The use of multiple cameras with overlapping fields of view enables robust tracking of people in highly crowded scenes. This may overshadow budget limitations when essential or sensitive areas are considered. The sharp decline in camera prices in recent years may further increase the feasibility of this setup.

Our main contribution is the use of multiple height homographies for head top detection, which makes our method robust to severe and persistent occlusions, and to challenging lighting conditions. Most of the false positives generated by this method are removed by a heuristic tracking scheme.

In the future we intend to investigate automatic setting of system parameters and to consider a distributed implementation of our algorithm. Another promising direction is to combine our algorithm with human body segmentation methods, to assist in false positive removal.

## Acknowledgments

This research was supported by the Israel Science Foundation (grant no. 1339/05). We would like to thank Ran Goldschmidt for assisting in data capture and in calibration and synchronization of the sequences.

## References

[1] Q. Cai and J.K. Aggarwal. Tracking human motion in structured environments using a distributed-camera system. *PAMI*, 21(11):1241–1247, 1999.

[2] G.K.M. Cheung, T. Kanade, J.Y. Bouguet, and M. Holler. A real time system for robust 3d voxel reconstruction of human motions. In *CVPR*, pages 714–720, 2000.

[3] O.D. Faugeras. *Three-Dimensional Computer Vision*. MIT Press, Boston MA, 1993.

[4] P. F. Felzenszwalb. Learning models for object recognition. In *CVPR*, pages 56–62, 2001.

[5] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. *PAMI*, 2007.

[6] J. S. Franco and Boyer E. Fusion of multi-view silhouette cues using a space occupancy grid. *ICCV*, 2:1747–1753, 2005.

[7] R. Goldschmidt and Y. Moses. Practical calibration and synchronization in a wide baseline multi-camera setup using blinking LEDs. Technical Report ftp://ftp.idc.ac.il/Pub/Users/cs/yael/TR-2008/IDC-CS-TR-200801, 2008.

[8] M. Isard and J. MacCormick. BraMBLe: a Bayesian multiple-blob tracker. In *ICCV*, pages 34–41, 2001.

[9] S.M. Khan and M. Shah. A multiview approach to tracking people in crowded scenes using a planar homography constraint. In *ECCV*, pages IV: 133–146, 2006.

[10] S.M. Khan, P. Yan, and M. Shah. A homographic framework for the fusion of multi-view silhouettes. In *ICCV*, 2007.

[11] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer. Multi-camera multi-person tracking for easyliving. In *International Workshop on Visual Surveillance*, 2000.

[12] A. Laurentini. The visual hull concept for silhouette-based image understanding. *PAMI*, 16(2):150–162, 1994.

[13] L. Lee, R. Romano, and G. Stein. Monitoring activities from multiple video streams: Establishing a common coordinate frame. *PAMI*, 22(8):758–767, 2000.

[14] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *CVPR*, volume 1, pages 878–885, 2005.

[15] A. Mittal and L. Davis. Unified multi-camera detection and tracking using region matching. In *Proc. of the IEEE Workshop on Multi-Object Tracking*, 2001.

[16] C. Papageorgiou and T. Poggio. Trainable pedestrian detection. In *IEEE Conference on Intelligent Vehicles*, pages 35–39, 1998.

[17] M. Quaritsch, M. Kreuzthaler, B. Rinner, Bischof. H., and B. Strobl. Autonomous multicamera tracking on embedded smart cameras. *Journal on Embedded Systems*, 2007.

[18] M. D. Rodriguez and M. Shah. Detecting and segmenting humans in crowded scenes. In *In Proc. Intern. Conf. on Multimedia*, pages 353–356, 2007.

[19] K Smith, D. Gatica-Perez, and J. Odobez. Using particles to track varying numbers of interacting people. In *CVPR*, pages 962–969, 2005.

[20] P. A. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *IJCV*, 63(2):153–161, 2005.

[21] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors. *IJCV*, 75(2):247–266, 2007.

[22] Q. Yu, G. Medioni, and I. Cohen. Multiple target tracking using spatio-temporal Markov Chain Monte Carlo data association. In *CVPR*, 2007.

[23] T. Zhao and R. Nevatia. Tracking multiple humans in complex situations. *PAMI*, 26(9):1208–1221, 2004.