Multi-camera Multi-object Tracking by Robust Hough-based Homography Projections

Sabine Sternig Thomas Mauthner Arnold Irschara Peter M. Roth
Horst Bischof
Institute for Computer Graphics and Vision
Graz University of Technology
{sternig,mauthner,irschara,pmroth,bischof}@icg.tugraz.at

Abstract

Recently, several approaches have been introduced for incorporating the information from multiple cameras to increase the robustness of tracking. This allows to handle problems of mutually occluding objects – a reasonable scenario for many tasks such as visual surveillance or sports analysis. However, these methods often ignore problems such as inaccurate geometric constraints and violated geometric assumptions, requiring complex methods to resolve the resulting errors. In this paper, we introduce a new multiple camera tracking approach that inherently avoids these problems. We build on the ideas of generalized Hough voting and extend it to the multiple camera domain. This offers the following advantages: we reduce the amount of data in voting and are robust to projection errors. Moreover, we show that using additional geometric information can help to train more specific classifiers drastically improving the tracking performance. We confirm these findings by comparing our approach to existing (multi-camera) tracking methods.

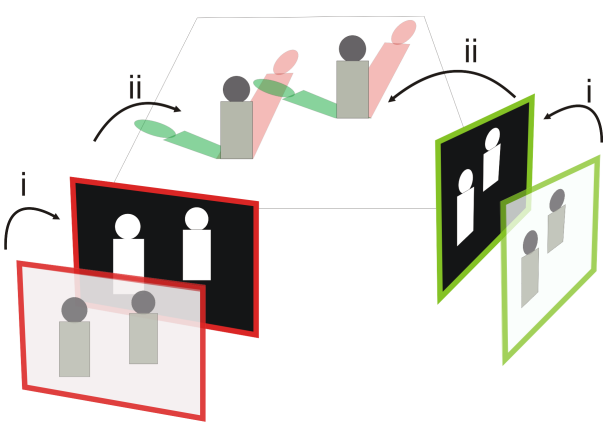
1. Introduction

Object tracking is an important task in computer vision and is often one of the first steps for video analysis in surveillance, sports, or industrial applications. Thus, there is still a high scientific interest and various successful methods have been proposed (*e.g.*, [7, 3, 2]). For multiple interacting objects occlusions make tracking very challenging. This especially applies for person tracking, when a large number of persons are occluding each other and the positions of single instances can no longer robustly be identified (*e.g.*, [6, 8, 12]).

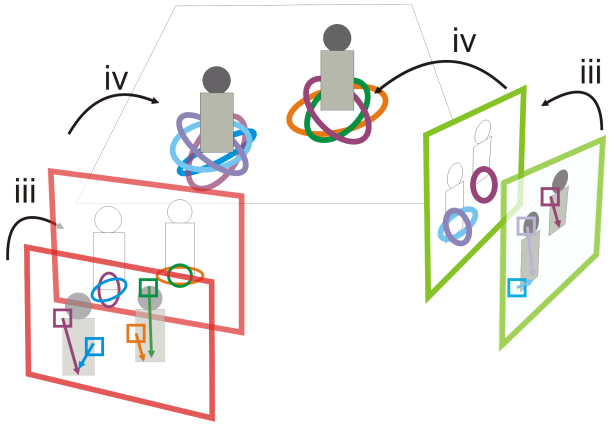
One way to overcome this problem is to take advantage of multiple cameras. Typical approaches for multi-camera tracking assume overlapping cameras observing the same 3D scene, exploiting several real-world constraints like a common geometry. One example for such real-world constraints for applications like person tracking are that objects are moving on a common ground-plane (*e.g.* [5, 13, 11, 20, 21]). In general, these methods first apply change detection [13, 20, 21] or a fixed pre-trained classifier [5] to estimate the foreground likelihood of specific pixels. Then, this information is fused exploiting the common ground plane by either estimating a score map [13, 20, 5] or by estimating axes intersections [21].

These methods, however, ignore several important issues hampering their applicability. First, detection and segmentation errors in the original views are projected onto the ground plane and have to be handled in the common view. Second, in general simple geometric transformations are only valid for a single ground plane, which results in unreliable projections for point not lying on the ground plane. Third, using a pixel-wise projection ignores imperfect localization in the different views and (minor) uncertainties in the homography. Altogether, this results in an inaccurate localization, making it hard to estimate an adequate back-projection onto the single view or could cause ghost projections (*i.e.*, a detection coming from the intersection of two unreliable projections) as shown in Figure 1(a).

To overcome these limitations, we introduce a new multiple camera tracking approach, which extends the ideas of generalized Hough voting [4] and implicitly deals with often ignored uncertainties in the projection. Thus, the contribution of the paper is twofold. First, we introduce a new Hough voting scheme which relates all foreground probabilities to a position on the ground plane. In this way the geometry information is preserved and the voting results can be fused over multiple cameras implicitly considering uncertainties in the projection and still preserving the beneficial properties such as robustness to occlusions. Second, using geometric verification and back-projection between views allows for adopting online to the scene for improv-



(a) Common approach to multi-camera tracking: based on background subtraction (i) the foreground pixels are projected to the common ground plane (ii), which may cause ghost detections and requires complex reasoning.



(b) Multi-camera tracking by joint Hough votes: foot-point voting to the ground plane generates a Hough map of each camera (iii), which are projected onto a common ground plane (iv) implicitly considering the geometric uncertainties.

Figure 1. Comparison between common approaches to multi-camera tracking based on background subtraction (a) and our proposed approach based on joint Hough voting (b) onto a common ground plane.

ing detection for each individual camera view, which further improves the tracking results over time. On top of the fused vote map we introduce a particle filtering approach exploiting geometric information to avoid overlapping particles on a top view map of the common ground plane.

The reminder of the paper is organized as follows. First, in Section 2 we summarize the main ideas of related homography-based multi-camera tracking approaches. Next, in Section 3 we introduce the novel multi-camera Hough tracking approach. Finally, we show experimental results in Section 4 and summarize and conclude the paper in Section 5.

2. Homographies and Tracking

In the following, we introduce the notation for homographies and give an overview of related tracking methods exploiting geometric information provided via a common ground-plane.

2.1. Homographies in Multiple Camera Setups

A general multi-camera tracking setup consists of n overlapping cameras, each of them observing the same 3D scene. Then each camera view v has its own local image coordinate system $\{x_v, y_v\}$ which can be mapped to the world coordinate system $\{X, Y, W\}$, requiring a fully calibrated setup. For a detailed review on single and multi-camera tracking we would refer to [20].

For many real-world tracking applications it can be assumed that the objects-of-interest are mainly moving on a common ground plane. Thus, the mapping of an image point $\mathbf{x}$ from a camera v onto a corresponding world point $\mathbf{X}$ on the ground plane can be realized by a plane to plane homography:

$$\mathbf{X} = \mathbf{H}_v \mathbf{x} \,, \tag{1}$$

where $\mathbf{H}_v$ is a homogeneous 3×3 matrix. Both, world and image coordinates, are given as homogeneous 3×1 vectors $\mathbf{X} = (X, Y, W)^\top$ and $\mathbf{x} = (x, y, 1)^\top$, respectively. The plane to plane homography defines the transformation up to scale, hence the matrices $\mathbf{H}_v$ have only 8 degrees of freedom. Real word positions can be computed by the normalization $\widetilde{\mathbf{X}} = (\widetilde{X}, \widetilde{Y}) = (X/W, Y/W)^\top$ [17].

2.2. Homography-based Tracking

Since homographies can easily be estimated (*e.g.*, by extracting SIFT points and running RANSAC), there has been a considerable interest for applying homography-based techniques for multi-camera tracking. Fleuret *et al.* [13] start with a simplified background subtraction then generate a generative model describing persons as rectangles. This is used to estimate a joint occupancy for each frame and for each position on the ground plane. By additionally using a color and a motion model the trajectories of multiple persons can be estimated. To avoid that the score map is polluted by other moving objects, Berclaz *et al.* [5] applied a detector instead of a simple background subtraction.

Khan and Shah [19] first obtain foreground likelihood maps for each view by applying a mixture of Gaussians model. These likelihood maps are then projected onto the ground plane using the given homographies and are accumulated into a synergy map. The synergy map is thresholded yielding the approximate feet positions of the persons, which are then back-projected onto each view. The actual

tracking is then performed using a look-ahead technique on the previously estimated foot-point positions in the ground plane. To make the tracking more robust, they slightly extended this approach [20] by sweeping over multiple planes parallel to the ground plane, to handle inaccurate projections.

Using the ground plane assumption for multi-camera tracking has two main disadvantages. First, the foot-points are often not visible due to occlusions and second, in different camera views the foot-points are not well defined (*e.g.*, frontal vs. side view). Thus, Eshel and Moses [10] [11] track the heads of persons. Moreover, similar to [20] they also sweep over different planes to capture persons of different heights. The main drawbacks of these approaches are that the heads must be visible in all views, that several planes have to be calibrated in parallel, and that the camera positions are limited to deep viewing angles.

A different approach to overcome the problem of inaccurately estimated foreground maps was proposed by Kim and Davis [21]. Starting with background subtraction they iteratively run a color segmentation step and estimate an increasingly better foreground map. The thus obtained foot-point might be inaccurate due to segmentation errors, they further propose to estimate the intersection of vertical axes of the estimated blobs to obtain a common localization in the top view. The actual tracking is then performed on the top view by applying a particle filter framework.

3. Multi-Camera Hough Tracking

When projecting image points $\mathbf{x}$ from perspective images to world points $\mathbf{X}$ onto the ground plane using the plane-to-plane homography $\mathbf{H}_v$ one has always to deal with uncertainty of these measures [9]. This uncertainty is influenced by two possible error sources, namely the uncertainty of the homography Σ_{H_v} and the uncertainty of the image point Σ_I resulting from the uncertainty in detection of the image point $\mathbf{x}$ in the image.

Following [17] the uncertainty Σ_X of a world coordinate $\mathbf{X}$, computed by the projection of an image point $\mathbf{x}$ using homography $\mathbf{H}_v$ is analytically given by

$$\Sigma_X = J_{H_v} \Sigma_{H_v} J_{H_v}^\top + J_I \Sigma_I J_I^\top \ , \tag{2}$$

where

$$J_I = \frac{\partial \mathbf{X}}{\partial \mathbf{x}} = \frac{1}{W} \begin{bmatrix} \boldsymbol{h}_1^\top & -X\mathbf{h}_3^\top \\ \boldsymbol{h}_2^\top & -Y\mathbf{h}_3^\top \end{bmatrix} \tag{3}$$

and

$$J_{H_v} = \frac{\partial \mathbf{X}}{\partial \mathbf{h}} = \frac{1}{W} \begin{bmatrix} \mathbf{x}^\top & 0 & -X\mathbf{x}^\top \\ 0 & \mathbf{x}^\top & -Y\mathbf{x}^\top \end{bmatrix} \tag{4}$$

are the Jacobian matrices, and $\mathbf{h}_i^\top$ is the i-th row of $\mathbf{H}_i$. Assuming that the correspondences for computation of the homography were accurately chosen, the uncertainty in Σ_{H_v} can be neglected. Thus, the uncertainty Σ_X at the ground plane position simplifies to

$$\Sigma_X = J_I \Sigma_I J_I^\top \ , \tag{5}$$

where the uncertainties in image coordinates

$$\Sigma_I = \begin{bmatrix} \sigma_{x^2} & \sigma_{xy} & 0 \\ \sigma_{xy} & \sigma_{y^2} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{6}$$

are derived from the inaccuracy in the image points.

However, these uncertainties are often ignored by existing multi-camera approaches (e.g., [13, 20, 21]). In the following, we introduce a multi-camera tracking approach building on the idea of generalized Hough voting [14, 23] that implicitly copes with these problems. We first run a detector and then, given the camera-to-ground plane homographies, we map the obtained votes onto the common top view map. This principle is depicted in Figure 1(b). Then, we introduce a multi-object particle filtering approach, which uses the prior knowledge that objects cannot occlude each other on a top view map. The tracking results can then be used to improve the combined vote maps by a novel view specific update scheme exploiting the geometric information to reduce the voting noise. In the following we will use the terms top view and ground plane interchangeable.

3.1. Multi-camera Hough Voting

To cope with projection errors we implicitly formulate the uncertainty in the world points Σ_X via Hough voting maps. In general, Hough Forests [14, 23] learn a mapping from image features onto a Hough space. Each Hough Forest $\mathcal{F}$ consists of a set of trees $\mathcal{T}$, where each tree $\mathcal{T}$ is constructed based on a set of patches $\mathcal{P}_i = (\mathcal{I}_i, c_i, \mathbf{d}_i)$; $\mathcal{I}_i$ is the appearance of the patch, c_i is the class label of the patch, and $\mathbf{d}_i$ is the offset vector of the patch with respect to the object's centroid.

During training, the class label uncertainty $U_1(\mathcal{P}) = |\mathcal{P}| \cdot Entropy(c_i)$ as well as the offset uncertainty $U_2(\mathcal{P}) = \sum_{i:c_i=1} (\mathbf{d}_i - \mathbf{d}_\mathcal{P})^2$ are optimized. The class label uncertainty enforces binary tests used as split criteria during the tree construction to consider the impurity of the class labels c_i, whereas the offset uncertainty enforces to group patches coming from a local environment. Finally, in a leaf node L the vote vectors $D_L = \{\mathbf{d}_i\}$ of the object patches and the foreground probability C_L are stored. During testing for a patch at position $\mathbf{y}$ the probability $p(E(\mathbf{x})|L(\mathbf{y}))$ is estimated, where $E(\mathbf{x})$ indicates whether an object is present at location $\mathbf{x}$ and $L(\mathbf{y})$ is the corresponding leaf node where the patch sampled at position $\mathbf{y}$ ends up. For each tree $\mathcal{T}$ the probability can be estimated as

$$p(E(\mathbf{x})|L(y)) = \begin{array}{l} p(\mathbf{d}(\mathbf{y}) = \mathbf{y} - \mathbf{x}|c(\mathbf{y}) = 1, L(\mathbf{y})) \\ \cdot p(c(\mathbf{y}) = 1|L(\mathbf{y})). \end{array} \quad (7)$$

The first term can be approximated by a Parzen window based on the offset vectors and the second term is the proportion of object patches C_L in a leaf node L at training time. The probabilities for each location $\mathbf{y}$ within the image are accumulated into a Hough map $\mathbf{V}$ over all trees $\mathcal{T}$ within the Hough forest. The actual detection task is finally performed by mode seeking in the thus obtained Hough map.

This idea can be extended for multiple camera views. In fact, we can generate a common Hough map, where the single view maps are accumulated by projecting them onto a common plane using the homographies. However, as described above such projections are prone to uncertainties Σ_X of a world point X resulting from the uncertainties of the corresponding image point $\mathbf{x}$. In our case an image point $\mathbf{x}$ is associated with a patch representation $\mathcal{I}_i(\mathbf{x})$ and the uncertainty results from inaccurately estimated endpoints of the vote vectors D_L, where L is the leaf node where $\mathcal{I}_i(\mathbf{x})$ ends up. The uncertainty Eq. (6) could be calculated over the endpoints of the vote vectors D_L within each leaf node. Alternatively, the uncertainty of the statistical distribution can also be approximated by Monte-Carlo simulation [17]. For our Hough voting scheme, however, this is already estimated within each of the leaf nodes L by the offset vectors D_L. Thus, we can implicitly handle the uncertainty in the projection to the common top view map.

The approach described so far builds on voting to the centroid of an object. Considering our intended application, *i.e.*, multi-camera object tracking, the centroid voting would require a large calibration effort. In fact, depending on the vote center (and therefore depending on the height of the person) different homographies would be required. Assuming that objects are moving on a common ground plane this large calibration effort can be avoided. Therefore, we modify the voting scheme: instead of voting for the objects centroid we propose to *vote for the foot-point* of the object. Since we know the plane to plane homography we can estimate the extend of the objects at all positions within the image. This information can be exploited to avoid a large evaluation effort by using the appropriate scale at different positions within the image.

Besides the implicitly handled uncertainties the joint multi-camera Hough voting enables the late fusion of detection information. Thus, all information is kept for tracking and we do not have to discard possibly useful information at a too early stage. Compared to approaches which solely rely on background subtracted images (as illustrated in Figure 1(a)) our detection based approach has further the advantage that we do not rely on motion and that only the object-of-interest is considered while other moving objects are ignored. The advantage of our voting concept is shown in the experimental section.

3.2. Multi-Camera Tracking

The common top view voting map $\mathbf{V}$, visualized in Figure 2, can now be used for multi-object tracking. Following the Hough voting scheme, we retrieve high votes on ground plane positions where an object-of-interest is localized. Although $\mathbf{V}$ does not express probabilities, it can be seen as a continuous confidence map. In contrast to existing single view approaches, the proposed ground plane Hough voting guarantees non-overlapping local maxima for each possible detection. This is guaranteed by the physical rule, that objects cannot overlap each other in the top view.

In particular, we use a particle filtering approach [18], which is widely used for tracking and provides a probabilistic framework for maintaining multiple hypotheses of the current object state. Particle filtering can be used to estimate the state of a system based on noisy measurements $\mathbf{z}$ by using a set of S weighted particles $\mathbf{x}^i_{1:k}, w^i_{1:k}$. In our case, given the set of S weighted particles $\{x^i_t, w^i_t\}$, $i = 0, ..., S$, at time step t we can estimate the probability distribution of the hidden target state $\mathbf{x}_t$ of the tracked object by $\mathbf{x}_t = [x, y, v_x, v_y]'$, where (x, y) are the center coordinates of the particles rectangle window and (v_x, v_y) are the velocities. The velocities are described by a Gaussian distribution with zero mean and motion dependent standard deviation, and each particle x^i_t simulates the real hidden state of the object. Using the dynamic model $p(x^i_t|x^i_{t-1})$ and the observation likelihood $p(z^i_t|x^i_t)$, the posterior distribution $p(\mathbf{x}_t|\mathbf{z}_t)$ is approximated by the finite set of particles $p(\mathbf{x}_t|\mathbf{z}_{1:t}) \approx \sum_{i=1}^{S} w^i_t x^i_t$.

The weights are updated according to

$$w^i_t \propto w^i_{t-1} \frac{p(z^i_t \mid x^i_t) p(x^i_t \mid x^i_{t-1})}{q(x^i_t \mid x^i_{t-1}, z^i_t)} , \quad (8)$$

where $\sum_{i=1}^{N_p} w^i_t = 1$ and $q(x^i_t \mid x^i_{t-1}, z^i_t)$ is the proposal distribution to draw particles from.

Using an auto-regression model, the transition probability $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ is represented by $\mathbf{x}_{t+1} = A\mathbf{x}_t + \mathbf{v}_t$. Applying the state transition model $p(x^i_t|x^i_{t-1})$ as proposal distribution leads to the bootstrap filter, where the weights are directly proportional to the observation model $p(z^i_t|x^i_t)$. Thus, we can define

$$w^i_t = w^i_{t-1} exp\left(log(\sum_{y \in x^i_t} G_b(y - x^i_t)) \right) , \quad (9)$$

describing the sum of the voting map V within a local neighborhood defined by x^i_t. G_B denotes the box filter approximation of a Gaussian kernel, which allows for the usage of

efficient integral image structures. Note that the sum has to be normalized between $[0, 1]$. Although the voting map cannot be seen as a probabilistic map, the particle filter is still working by seeking for the strongest local mode. Finally, the posterior density $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ is approximated by the weighted mean over the particle distribution, as given in Eq. (8). To avoid the degeneracy of the particle set, the re-sampling of the weights is performed after each frame. For more details on particle filtering we would refer to [1].

The foot-point voting on the centralized top view map enables us to exploit the knowledge that objects cannot overlap each other on the top view (see Figure1(b)) and to incorporate this to the particle filter framework. So far each object is tracked by an individual particle filter, without any knowledge about surrounding objects. Each object o has its own particles $\mathbf{x}_t^{i,o}$, with $i = 1, ...S_o$, which are re-weighted according to Eq. (9). In general, this leads to hijacked particles, where several trackers are following the same voting maxima, which is often called the "error merge" problem.

After re-weighting the particles for each individual object o, we introduce a joint re-weighting, where particles of different objects $\mathbf{x}_t^{i,o1}$ and $\mathbf{x}_t^{j,o2}$ are penalized if they are overlapping [22]. Penalizing such overlapping particles avoids that particles belonging to different objects merge to one maximum within the common vote map. This can be seen related to the magnetic-inertia potential model [24], which proposes to model a gravitation and magnetic repulsion scheme. But, in contrast to [24] the non-overlapping assumption is directly assured in our concept as a result of the ground plane projections described in Section 3.1.

3.3. View-specific Hough Voting

Random forests (and therefore also Hough forests) are a perfect choice for learning generic classifiers, since they allow for training from huge data sets and can cope with multi-modal data. However, for specific camera views not all information is needed and the large variability in the data would cause some noise in the Hough votes increasing the uncertainty of the world points Σ_X. One way to overcome these problems would be to train a separate Hough forest for each camera view, however, requiring a massive labeling effort.

In contrast, we exploit the geometric constraints by using the back-projection of the tracking results on the top view map to each of the camera views. In this way, we can adapt a general pre-trained classifier to every camera in order to reduce the amount of noise. We introduce an additional view specific term $p(P_v|c = 1, L(\mathbf{y}))$ for each vote vector $\mathbf{d}_i$ in the leaf node L, where P_v considers only object patches that vote correctly within this specific view. To reduce the importance of a vote vector $\mathbf{d}_i$ which is not suitable for a specific view v and hence introduces noise, we are now interested in $p(E_v(\mathbf{x})|L(\mathbf{y}))$, where $E_v(\mathbf{x})$ is the evidence of an object at location $\mathbf{x}$ in camera view v. By approximating $p(\mathbf{d}(\mathbf{y}) = \mathbf{y} - \mathbf{x}|c(\mathbf{y}) = 1, L(\mathbf{y}))$ by a sum of Dirac measures $\delta_{\mathbf{d}_i}(\mathbf{y} - \mathbf{x})$, as shown in [15], the view specific probability can now be calculated as

$$p(E_V(\mathbf{x})|L(y)) = \frac{1}{|\mathcal{P}_{L(y)}|} \cdot \sum_{P_v \in \mathcal{P}_{L(y)}} p(P_v|c = 1, L(\mathbf{y})) \cdot p(c = 1|L(\mathbf{y})) \cdot \delta_{\mathbf{d}_i}(\mathbf{y} - \mathbf{x}). \tag{10}$$

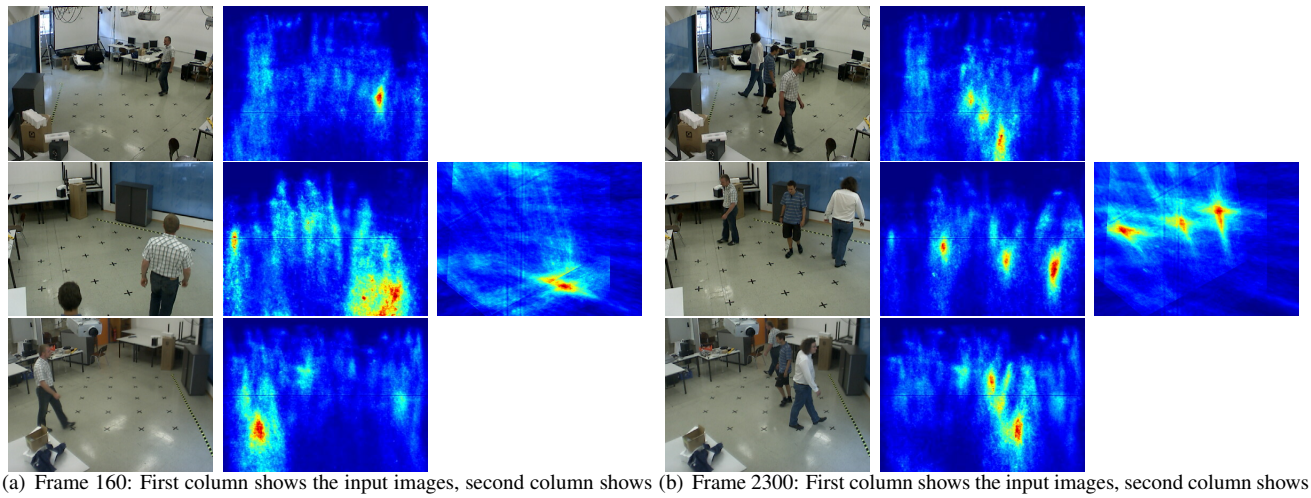
The view specific term $p(P_v|c = 1, L(\mathbf{y}))$ is updated over time by using the back-projection of the tracking results on the common ground plane. Therefore, we count how often a specific vote $\mathbf{d}_i$ votes into a correct position (given by the back-projection of the tracking results coming from the particle filter) and denote this number by $n^+_{\mathbf{d}_i}$. In addition, we count how often this vote casts to a wrong location, *i.e.*, where no object-of-interest is present: $n^-_{\mathbf{d}_i}$. The view specific term can now be calculated by

$$p(P_i \in V|c = 1, L(\mathbf{y})) = \begin{cases} 0.5 & \text{if } sum_n = 0 \\ \dfrac{n^+_{\mathbf{d}_i}}{n^+_{\mathbf{d}_i} + n^-_{\mathbf{d}_i}} & \text{otherwise} , \end{cases} \tag{11}$$

where $sum_n = n^+_{\mathbf{d}_i} + n^-_{\mathbf{d}_i}$. The benefits of the view-specific updates are illustrated in Figure 2, where we show the evolution of the vote maps for each camera over time. Figure 2(a) shows the vote maps for each of the three camera views for frame 160, where it can be seen that a lot of noisy votes are reported for the background. In contrast, for frame 2300 shown in Figure 2(b), the level of noise is decreased in the view specific vote maps as well as in the combined top view vote map, which demonstrates the effect of view-specific updates.

4. Experiments

In the following, we demonstrate our approach on different publicly available datasets for multi-camera object tracking. For the experiments, we trained a Hough forest [14] voting to the foot-point of the object on the VIPeR pedestrian data set [16]. The forest consists of three trees, each of it having a maximum depth of 15. In addition, we give a comparison to two baseline approaches. First, to a background subtraction (BGS) based approach, where we project the foreground pixels of each camera onto the common ground plane to obtain a summed common foreground pixel top view map. Second, to a single camera approach, which builds on the same Hough maps as the proposed approach. For all approaches we apply the same particle filter framework holding a set of 300 particles described in Section 3.2, where the tracking is initialized manually, but the probability maps are derived in a different way.



(a) Frame 160: First column shows the input images, second column shows the corresponding Hough maps of each camera view and the third column shows the projected common Hough map of the top view.

(b) Frame 2300: First column shows the input images, second column shows the corresponding Hough maps of each camera view and the third column shows the projected common Hough map of the top view.

Figure 2. View-specific Hough votes: Input images, Hough maps of individual views and common top view Hough map. The evolution of Hough maps over time – showing the maps for frames 160 and frames 2300 demonstrates the effect of the updates. The single votes contain less noise resulting in much better combined top-view maps.

For the first comparison the probability maps come from a simple background subtraction. In this case the multiple-camera information is exploited. For the second comparison no multiple-camera information is used but the probability maps build on the original Hough maps.

The first experiment we run on the *Set 1* sequence of the publicly available Medium Dataset [25]. The dataset, showing an indoor lab environment, captures three people walking around. The persons occlude each other and are captured by three different cameras. Each video contains about 2500 frames with a resolution of 384×288 pixels. To give a quantitative evaluation we annotated every tenth frame and estimated the pixel error on the ground plane. The corresponding results over time are shown in Figure 3.

It can be seen that at the beginning both multi-camera approaches yield a comparable performance, but after the persons move too close to each other (which happens around frame 1000) the quality of the background subtraction based approach is decreasing. The same can also be recognized for the single view trackers, however, here the tracking accuracy is degraded much earlier. In particular, these methods are suffering from the "error merge" problem as well as the "labeling problem. The first one, which especially applies for the single view trackers, describes the problem that the tracker looses its specific instance and falsely coalesces with others. The second one means that identities of the objects are mixed up by the trackers. However, it can be seen that the proposed approach avoids both problems and thus yields much more stable tracking results. Additionally, we give the averaged pixel error for all approaches in Table 1 and show illustrative results in Figure 4.

In the second experiment, we run the same setup as described above on the publicly available *Campus Sequence 2* [5] consisting of 5884 images from three cameras with a resolution of 360×288 pixels showing an outdoor sequence with three moving persons. The obtained error rates averaged over the whole sequence are listed in Table 1. In general, considering the error rates it is revealed that this scenario of lower complexity than the other one – a larger area is observed and the number of occlusions is smaller. This also explains the rather good results for the simple background subtraction based approach. However, as for the previous setup it can be seen that using the combined

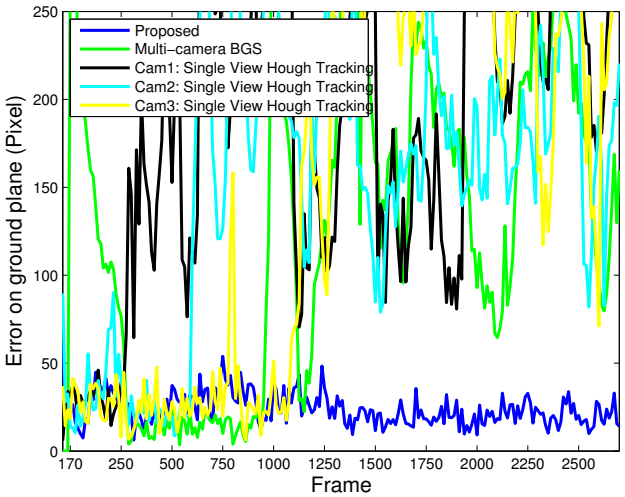


Figure 3. Medium *Set 1* sequence: Error in pixel on the ground plane stays constant over time for the proposed approach, but increases for all other approaches.

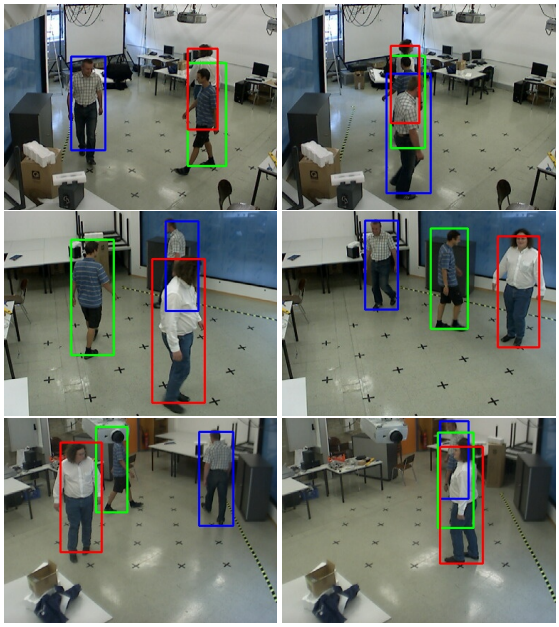


Figure 4. Medium *Set 1* sequence: Illustrative tracking results.

multi-camera approach the tracking results can significantly be improved. Even though we do not use an instance specific tracking approach the view-specific updates, which reduce the noise within the common vote map, in combination with the non-overlapping constraint of the particle filter enables our approach to track both sequence without any error merge or labeling problems. Finally, illustrative results for this data set are shown in Figure 5.

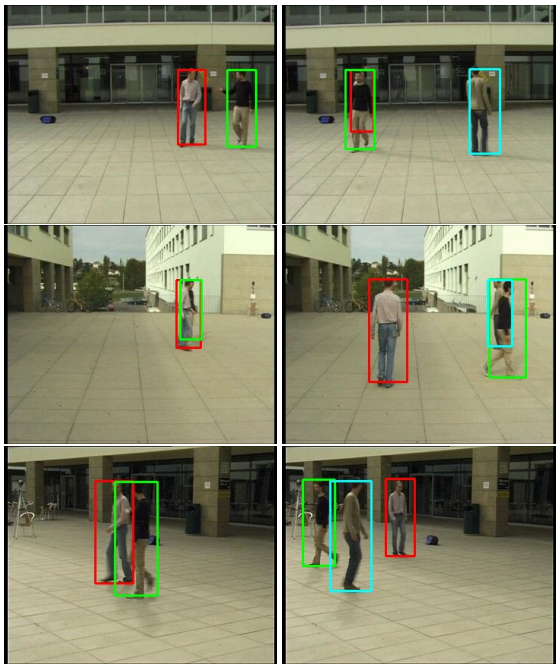


Figure 5. Illustrative results on the *Campus Sequence 2*.

Approach	Set 1	Campus 2
Proposed	**23.9**	**15.1**
BGS Based	128.2	27.2
HV Cam1	186.8	80.8
HV Cam2	153.8	77.5
HV Cam3	152.6	136.0

Table 1. Comparison of mean error in pixels on the top view map for *Set 1* and *Campus 2* sequences.

Additionally, we evaluate our approach on the Apidis dataset[1], showing a basket ball game from 7 cameras. We use camera 1, 2, 4 and 7 for our evaluation. To reduce computational complexity we use an image size of 400×300 pixels. Illustrative results are shown in Figure 6.

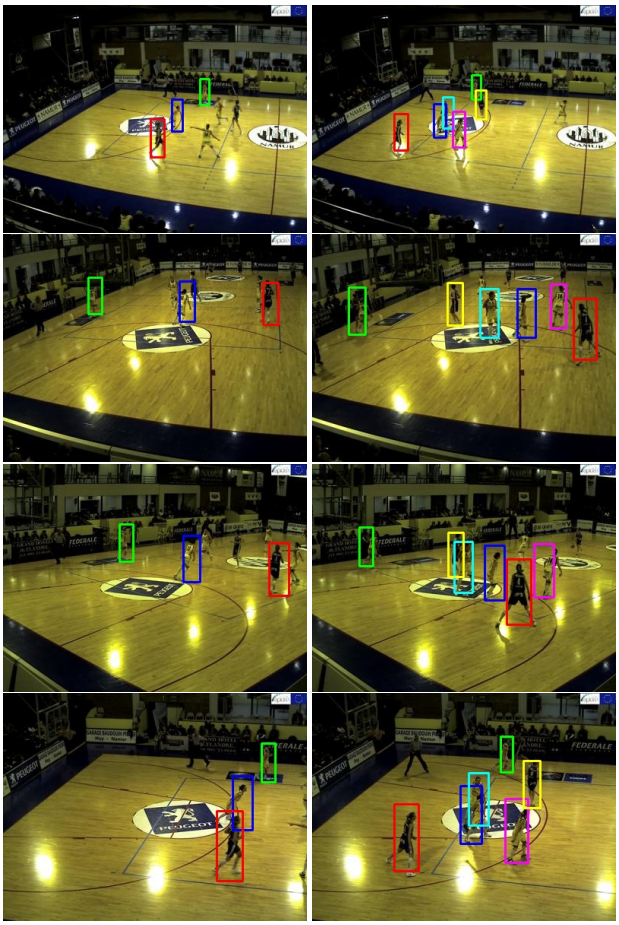


Figure 6. Illustrative results on the *Apidis* for camera 1,2,4 and 7, covering one half of the playground.

5. Conclusion

Most multiple camera approaches for multiple object tracking rely on background subtraction and project all foreground pixels onto a common ground plane. Hence, hurt-

[1] http://www.apidis.org/Dataset/

ing the geometric constraints large projection errors are introduced resulting in ghost detections. To overcome these limitations, we propose a novel multi-camera tracking approach, where we introduce a multi-camera Hough voting scheme. The key idea is to direct the votes to the foot-points instead of to the centroid. In this way, exploiting geometric constraints we can map the single camera votes onto a common ground plane and can implicitly handle geometric uncertainties. Additionally, we use an extended more robust particle filtering tracking approach, where the constraints given by the common ground-plane are exploited, *i.e.*, that objects cannot occupy the same position at the same time. Having the positions of the tracked objects we back-project the tracking results onto each individual view to identify instable votes. This further allows us to perform view specific updates, reducing the noise within each individual Hough map. Overall, we get a robust multiple object tracking approach, which avoids the error merge and labeling problem, even though no instance specific information is used. Future work will concentrate on automatic initialization and canceling of the object tracks.

Acknowledgement

This work was supported by the Austrian Science Fund (FWF) under the project MASA (P22299) and by the Austrian Research Promotion Agency (FFG) under the projects MobiTrick (8258408) and HOLISTIC (830044) under the FIT-IT programme and the FFG project SECRET (821690) under the Austrian Security Research Programme KIRAS.

References

[1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 2002.

[2] S. Avidan. Ensemble tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.

[3] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33(7):1324–1338, 2011.

[4] D. H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 1981.

[5] J. Berclaz, F. Fleuret, and P. Fua. Principled detection-by-classification from multiple views. In *Proc. Int'l Conf. on Computer Vision Theory and Applications*, 2008.

[6] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. v. Gool. Online multi-person tracking-by-detection from a single, uncalibrated camera. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2010.

[7] W. Brendel, M. Amer, and S. Todorovic. Multiobject tracking as maximum weight independent set. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2011.

[8] Y. Cai, N. de Freitas, and J. J. Little. Robust visual tracking for multiple targets. In *Proc. European Conf. on Computer Vision*, 2006.

[9] A. Criminisi, I. Reid, and A. Zisserman. A plane measuring device. In *Proc. British Machine Vision Conf.*, 1997.

[10] R. Eshel and Y. Moses. Homography based multiple camera detection and tracking of people in a dense crowd. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[11] R. Eshel and Y. Moses. Tracking in a dense crowd using multiple cameras. *Int'l Journal of Computer Vision*, 2010.

[12] A. Ess, B. Leibe, K. Schindler, and L. v. Gool. On-line adaption of class-specific codebooks for instance tracking. In *Proc. British Machine Vision Conf.*, 2010.

[13] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2008.

[14] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[15] J. Gall, N. Razavi, and L. J. v. Gool. On-line adaption of class-specific codebooks for instance tracking. In *Proc. British Machine Vision Conf.*, 2010.

[16] D. Gray, S. Brennan, and H. Tao. Evaluating appearance models for recognition, reacquisition, and tracking. In *Proc. IEEE Workshop on Performance Evaluation of Tracking and Surveillance*, 2007.

[17] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[18] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. European Conf. on Computer Vision*, 1996.

[19] S. M. Khan and M. Shah. A multiview approach to tracking people in crowded scenes using a planar homography constraint. In *Proc. European Conf. on Computer Vision*, 2006.

[20] S. M. Khan and M. Shah. Tracking multiple occluding people by localizing on multiple scene planes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2009.

[21] K. Kim and L. Davis. Multi-camera tracking and segmentation of occluded people on ground plane using search-guided particle filtering. In *Proc. European Conf. on Computer Vision*, 2006.

[22] T. Mauthner and H. Bischof. A robust multiple object tracking for sport applications. In *Proc. Workshop of the Austrian Association for Pattern Recognition*, 2007.

[23] R. Okada. Discriminative generalized hough transform for object dectection. In *Proc. IEEE Int'l Conf. on Computer Vision*, 2009.

[24] W. Qu, D. Schonfeld, and M. Mohamed. Real-time interactively distributed multi-object tracking using a magnetic-inertia potential model. In *Proc. IEEE Int'l Conf. on Computer Vision*, 2005.

[25] P. M. Roth, C. Leistner, A. Berger, and H. Bischof. Multiple instance learning from multiple cameras. In *Proc. IEEE Workshop on Camera Networks*, 2010.