Tracking a Large Number of Objects from Multiple Views

Zheng Wu[1], Nickolay I. Hristov[2], Tyson L. Hedrick[3], Thomas H. Kunz[2], Margrit Betke[1*]
[1] Department of Computer Science, Boston University
[2] Department of Biology, Boston University
[3] Department of Biology, University of North Carolina at Chapel Hill

Abstract

We propose a multi-object multi-camera framework for tracking large numbers of tightly-spaced objects that rapidly move in three dimensions. We formulate the problem of finding correspondences across multiple views as a multidimensional assignment problem and use a greedy randomized adaptive search procedure to solve this NP-hard problem efficiently. To account for occlusions, we relax the one-to-one constraint that one measurement corresponds to one object and iteratively solve the relaxed assignment problem. After correspondences are established, object trajectories are estimated by stereoscopic reconstruction using an epipolar-neighborhood search. We embedded our method into a tracker-to-tracker multi-view fusion system that not only obtains the three-dimensional trajectories of closely-moving objects but also accurately settles track uncertainties that could not be resolved from single views due to occlusion. We conducted experiments to validate our greedy assignment procedure and our technique to recover from occlusions. We successfully track hundreds of flying bats and provide an analysis of their group behavior based on 150 reconstructed 3D trajectories.

1. Introduction

The interpretation of the motion of large groups of individuals is a difficult problem in computer vision. A complete tracking system typically consists of two phases: estimation of the state of each object and *across-time* data association (i.e., the assignment of current measurements to object tracks). State estimation is difficult when object motion is not smooth; data association is difficult when the population of objects is dense. This paper stresses the latter scenario in a multi-view setting. This means we also need to consider an *across-view* data association problem: the determination of corresponding measurements in multiple views. Tracking is challenging here because it involves solving the problem of matching hundreds of detected individuals from frame to frame and from camera view to camera view and reasoning about their occlusions.

Past efforts have incorporated models of the occlusion process [15] or the interaction of individuals [12, 22], knowledge about the appearance of the objects [14, 10] or the homography of the scene [11, 8], or have applied trajectory relinking [21, 16, 23]. The tracking scenarios that have been considered in the past have typically involved interpreting the activities of fewer than ten individuals per image frame. Earlier methods typically do not scale well in cases when there are hundreds of objects moving in three-dimensional (3D) space and where objects differ by only a few visual cues. Our work, on the other hand, falls in the category of recent research efforts to understand the interaction of significantly larger crowds of individuals [1, 2, 5, 6, 17]. Our contributions are:

- A new formulation for across-view data-association in large crowds using a likelihood function that is based on multi-view geometry.
- A new iterative search procedure (IGRASP) to solve the across-view data-association problem.
- A stereoscopic method to reconstruct the trajectories of objects moving in 3D space that employs a new epipolar-neighborhood search.
- A new information fusion technique that ensures interpretation of occlusion and consistency of tracking.

We formulate the problem of finding object correspondences across multiple views as a multidimensional assignment problem. This problem is known to be NP-hard, but there are suboptimal algorithms that can determine assignments efficiently. To handle scenarios where objects occlude each other, we modified a greedy randomized adaptive search algorithm [18] that does not adhere to the traditional one-to-one correspondence assumption that each object is represented by one measurement. After establishing correspondences between the measurements in each view and the objects that are being tracked, our method computes the 3D trajectories of the objects via stereoscopic reconstruction. The accuracy of tracking in each camera view

*This material is based upon work supported by the National Science Foundation under Grant Nos. 0326483 and 0910908.

2009 IEEE 12th International Conference on Computer Vision (ICCV)
978-1-4244-4419-9/09/$25.00 ©2009 IEEE

was improved by examining the consistency of tracker-to-tracker associations. We incorporated our algorithms into a tracking system that can successfully reason about the movements of hundreds of individuals recorded from multiple views.

In single-view tracking, ambiguity caused by occlusion can be solved by optimizing some global function that considers trajectory smoothness over several frames. With this approach, trajectory pieces ("tracklets") are linked successfully and full trajectories can be recovered (e.g., [21, 16, 23]). The approach assumes that the occlusion will disappear within the typical tracking period. However, this assumption does not hold in situations when hundreds of objects emerge at the same time in the scene and occlusion occurs constantly, and for these situations, single-view approaches are not promising.

An alternative way is using more than one camera to provide tracking information from different views [13]. Most of the previous multi-view works on tracking pedestrians use homography [8, 11] as a natural and effective approach to find the correspondence across different views. Occlusion can then be resolved even if the object is completely occluded in some views. Extending homography-based approaches to the case when objects are not moving in the plane, as in our case, is not intuitive.

We stress the difficulties of our tracking problem: The objects are not easily distinguishable based on appearance, and, with a large number of objects moving in 3D space, occlusion frequently occurs. This problem is relevant for the analysis of group behavior of animals [5, 12, 19], the application we chose in this paper, and for trajectory-based abnormality detection in surveillance studies [1, 2, 6, 20]. The results of surveillance or animal-analysis systems usually depend on the trajectories that the tracker produced. Our tracking system can therefore have an impact in these applications when it uses, as a post processing step, the same approaches to trajectory analysis. Our experiments show not just the effectiveness of our tracking system, but also provide information valuable to mammalogists, ecologist, and conservation biologists. In particular, we produced the first stereoscopic analysis of the emergence behavior of free-ranging bats. We report the first accurate and reproducible estimates of 3D velocities of groups of emerging bats and their spatio-temporal interactions.

2. Multi-object Multi-view Tracking

We first describe our multi-object tracking approach and formulate the multi-view data-association problem (Sec. 2.1). We then introduce an iterative search procedure to efficiently solve this NP-hard problem (Sec. 2.3). We use stereoscopic reconstruction to combine the two-dimensional trajectories from each view into a single three-dimensional trajectory for each object and introduce the technique "epipolar-neighborhood search" (Sec. 2.2). We explain how we ensure the consistency of this "sensor fusion" in the presence of occlusions (Sec. 2.4). The architecture of our tracking system is shown in Fig. 1.

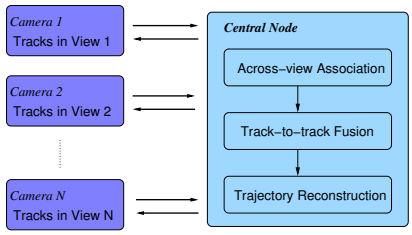

Figure 1. The hybrid architecture of our tracking system. Tracking is performed at each sensor level and tracks and measurements are sent to a central node for processing. Each sensor tracker adjusts its across-time associations based on the fusion result it receives from the central node.

2.1. Multidimensional Assignment Formulation

In this section, we describe how we adapted the recursive Bayesian techniques from the radar literature [4, 7] to address the across-camera data association problem. Our contribution includes a formulation of the likelihood function that is based on multi-view geometry. The function determines how likely it is that associated 2D measurements are projections of an object in the 3D scene.

Given N calibrated and synchronized cameras that share overlapping fields of view and n_s measurements in the field of view of camera s, the state $x^{(t)}$ (3D coordinates) of an object of interest at time t can be assumed to evolve in time according to the equations

$$x^{(t+1)} = Ax^{(t)} + v^{(t)} \quad (1)$$

as observed via measurements

$$z^{(t)}_{s,i_s} = H_s\, x^{(t)} + w^{(t_s)} \text{ for } s = 1, ..., N,\ i_s = 1, ..., n_s, \quad (2)$$

where $v^{(t)}$ and $w^{(t_s)}$ are independent zero-mean Gaussian noise processes with respective covariances $Q(t)$ and $R_s(t)$, A is the state transition matrix, and H_s the projection matrix for camera s. Each measurement $z^{(t)}_{s,i_s}$ is either the projected image of some object a in camera s plus additive Gaussian noise $\mathcal{N}(0, R_s(t))$, or a false-positive detection, which is assumed to occur uniformly within the field of view of camera s. For each camera, the detection rate is $\mathrm{P}_{D_s} < 1$. We add "dummy" measurements $z^{(t)}_{s,0}$ to handle the case of missed detections. In particular, when object a is not detected in camera s at time t, dummy measurement $z^{(t)}_{s,0}$ from camera s is associated with object a.

For ease of notation, we now drop the superscript t. We use the notation $Z_{i_1 i_2 \ldots i_N}$ to indicate that the measurements $z_{1,i_1}, z_{2,i_2}, \ldots, z_{N,i_N}$ originated from a common object in

the scene at time t. The likelihood that $Z_{i_1 i_2 ... i_N}$ describes object state x_a is given as

$$p(Z_{i_1 i_2 ... i_N}|x_a) = \prod_{s=1}^{N} \{[1 - \mathrm{P}_{D_s}]^{1-u(i_s)} \times [\mathrm{P}_{D_s}\, p(z_{s,i_s}|x_a)]^{u(i_s)}\} \quad (3)$$

where $u(i_s)$ is an indicator function defined as

$$u(i_s) = \begin{cases} 0 & \text{if } i_s = 0 \\ 1 & \text{otherwise,} \end{cases} \quad (4)$$

and the conditional probability density of a measurement z_{s,i_s}, given it originated from object a, is

$$p(z_{s,i_s}|x_a) = \mathcal{N}(z_{s,i_s}; H_s\, x_a, R_s). \quad (5)$$

The likelihood that $Z_{i_1 i_2 ... i_N}$ is unrelated to object a or related to dummy object $\oslash$ is

$$p(Z_{i_1 i_2 ... i_N}|\oslash) = \prod_{s=1}^{N} [\frac{1}{\Phi_s}]^{u(i_s)}, \quad (6)$$

where Φ_s is the volume of the field of view of camera s. Since we do not know the true state x_a in Eq. 5, we replace it by

$$\hat{x}_a = \arg\min_{x_a} \sum_{s=1}^{n} d(z_{s,i_s}, H_s\, x_a), \quad (7)$$

where d is Euclidean distance between $H_s\, x_a$, the object position projected onto the image plane s, and the corresponding measurement z_{s,i_s}. Using stereoscopy,[1] we estimate the state $\hat{x}_a$ to be the reconstructed 3D position based on the corresponding measurements $z_{1,i_1}, z_{2,i_2}, ..., z_{N,i_N}$ in the N views. We now can define the cost of associating N-tuple $Z^t_{i_1 i_2 ... i_N}$ to object a at time t is as the negative log-likelihood ratio: [2]

$$\begin{aligned} c_{i_1 i_2 ... i_N} &= -\ln \frac{p\,(Z_{i_1 i_2 ... i_N}|\, a)}{p\,(Z^t_{i_1 i_2 ... i_N}|\oslash)} \\ &= \sum_{s=1}^{N} \{[u(i_s) - 1] \ln(1 - \mathrm{P}_{D_s}) \\ &\quad - u(i_s) \ln \left(\frac{\mathrm{P}_{D_s} \Phi_s}{|2\pi R_s|^{1/2}} \right) \\ &\quad + u(i_s)[\frac{1}{2}(z_{s,i_s} - H_s \hat{x}_a)^T R_s^{-1} (z_{s,i_s} - H_s \hat{x}_a)]\} \end{aligned} \quad (8)$$

We use binary variable $x_{i_1 i_2 ... i_N}$ to indicate if $Z_{i_1 i_2 ... i_N}$ is associated with a candidate object or not. Assuming that such associations are independent, our goal is to find the most likely set of n-tuples that minimizes the linear cost function

$$\begin{aligned} c &= \min \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} ... \sum_{i_N=0}^{n_N} c_{i_1 i_2 ... i_N}\, x_{i_1 i_2 ... i_N} \quad (9) \\ \text{s. t.} \quad & \sum_{i_2=0}^{n_2} \sum_{i_3=0}^{n_3} ... \sum_{i_N=0}^{n_N} x_{i_1 i_2 ... i_N} = 1; \quad i_1 = 1, 2, ..., n_1 \\ & \sum_{i_1=0}^{n_1} \sum_{i_3=0}^{n_3} ... \sum_{i_N=0}^{n_N} x_{i_1 i_2 ... i_N} = 1; \quad i_2 = 1, 2, ..., n_2 \\ & \vdots \\ & \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} ... \sum_{i_{N-1}=0}^{n_{N-1}} x_{i_1 i_2 ... i_N} = 1; \quad i_N = 1, 2, ..., n_N. \end{aligned}$$

Eq. 9 is known as the multidimensional assignment problem, which it is NP-hard for the dimension $N \geq 3$. The processing time for the optimal solution is unacceptable in dense tracking scenarios, even if a branch-and-bound search method is used, because such a method is inevitably enumerative in nature. The alternative is to search for a sub-optimal solution to this combinatorial problem, using greedy approaches and its variants, Lagrangian relaxation, simulated annealing or tabu search. We choose the Greedy Randomized Adaptive Search Procedure (GRASP) [18] as the basic paradigm and modified it to handle occlusion reasoning (Sec. 2.3).

2.2. Generic GRASP in Multi-view Scenario

We briefly outline a generic GRASP implementation for the multidimensional assignment problem [18] and then adjust it to our multi-view scenario:

GREEDY RANDOMIZED ADAPTIVE SEARCH PROCEDURE:

Initialization by computing the costs for all possible associations

For $i = 1, ..., maxIter$

1. Randomly construct a feasible greedy solution,
2. Recursively improve the feasible solution by local search,
3. Update the best solution by comparing the total costs,

Output the best solution found so far.

In the local search phase, we adopt the so-called 2-assignment-exchange operation. That is, for two tuples $Z_{i_1 ... i_j ... i_N}$ and $Z_{i'_1 ... i'_j ... i'_N}$ from the feasible solution, we exchange the assignment to $Z_{i_1 ... i'_j ... i_N}$ and $Z_{i'_1 ... i_j ... i'_N}$ if such operation decreases the total cost. The exchange takes place recursively until no exchange can be made anymore.

We adopt a technique similar to "gating" during the initialization step to reduce the number of possible candidate tuples as follows. Given a pair of calibrated views, our

[1] We selected the Direct Linear Transformation (DLT) algorithm [9] to perform the 3D reconstruction because of its efficiency and sufficient accuracy. Other methods may replace DLT in our framework.

[2] We can append to this cost function other types of costs, e.g., the measures of object appearance, if such measures are available, and define a reasonable weighing scheme to yield normalization.

technique establishes the correspondence of the two projected images of an object using epipolar geometry. Thus, we only need to evaluate the candidate tuples that lie within the neighborhood of corresponding epipolar lines. To enforce this neighborhood search, we set the cost of associating measurements that violate the epipolar-geometry constraints to a large number. This pruning step in building the multidimensional assignment problem, which we call epipolar-neighborhood search, becomes crucial for the overall efficiency, which will be demonstrated in Sec. 3.

2.3. Iterative GRASP in Multi-view Scenario

The constraints in Eq. 9 imply the one-to-one correspondence between measurements and objects, except for the dummy measurement and its corresponding object. Each measurement is either assigned to some object or claimed to be a false-positive detection. An object is either measured in each view or it is missed. This strict formulation is not desirable in the multi-view tracking scenario, as shown in Fig. 2. With the one-to-one correspondence constraint, the numeric optimal solution might associate $(z_{1,1}, z_{2,1})$ to object o_1 and $(z_{1,3}, z_{2,2})$ to object o_2 or decide object o_2 is not detected in view 1. This ambiguity is difficult to resolve since both interpretations have acceptable total costs. Our basic assumption is that if an occlusion occurs in one view, it does not happen in other views at the same time. This requires that we relax the one-to-one correspondence constraint: Measurements that overlap due to occlusion or imperfect segmentation during the detection stage and thus are interpreted as a single measurement can be assigned to multiple objects.

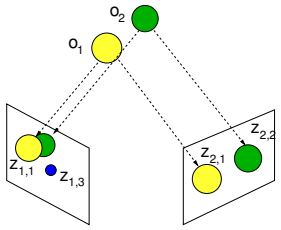

Figure 2. Stereoscopic reasoning for assessing occlusion. From a single view, two objects o_1 and o_2 occlude each other and yield a single measurement $z_{1,1}$. A single-view tracker may lose track of one of the objects or may misinterpret the nearby false-positive detection $z_{1,3}$ as one of the objects. If two views are available, the objects o_1 and o_2 can be matched to their respective measurements $z_{2,1}$ and $z_{2,2}$. Stereoscopic reasoning reveals that $z_{1,1}$ is the image of both objects and $z_{1,3}$ an unrelated measurement.

We denote the set of all possible N-tuples as $F = Z_1 \times ... \times Z_i \times ... \times Z_N$, where Z_i is the set of all the measurements in view i plus the "dummy" measurement. Solving Eq. 9 yields a set of assignments for the N-image measurement set Z, where a specific assignment can be written as $\{z_{i_1 i_2 ... i_N} | x_{i_1 i_2 ... i_N} = 1\}$. We divide the set of assignments into two subsets as follows:

1. **Confirmed associations:** $M_c = \{Z_{i_1 i_2 ... i_N} | x_{i_1 i_2 ... i_N} = 1; i_1 \neq 0; ...; i_N \neq 0\}$.
2. **Suspicious associations:** $M_s = Z \setminus M_c$.

Suspicious associations involve both dummy measurements $z_{s,0}$ that indicate an object was not detected in some view and measurements that were assigned to the dummy object $\oslash$ (i.e., false positive detections).

Eq. 9 does not contain constraints with zeros for index i. Associations in set M_s have at least one zero in their subscripts.

The new version of GRASP that we propose here (see pseudocode for Iterative GRASP below) computes a solution to an assignment problem that is described by Eq. 9, except with the already confirmed assignments in M_c removed from the feasible assignment set F. During an iteration of Iterative GRASP, an assignment found greedily in the construction phase can thus not involve a tuple already in M_c. The algorithm generates two subsets from the resulting solution and iterates until a maximum number of iteration is reached or M_c in the current iteration is empty.

ITERATIVE GREEDY RANDOMIZED ADAPTIVE SEARCH PROCEDURE (IGRASP):

Building Phase

Initialization by computing the costs for all possible associations in set F;

Solving Phase

For $i = 1, ..., maxIter$

1. Formulate multidimensional assignment problem on set F,
2. Run standard GRASP described in Sec. 2.2,
3. Partition the computed solution into confirmed set M_c and suspicious set M_s.
4. **If** Set M_c is empty, terminate; **else** $F = F \setminus M_c$

Output the best solution found so far.

2.4. Multi-Object Tracking with Fusion of Information from Multiple Views

Thus far we described a method to solve multi-view data association in a single time step. The resulting solution allows us to estimate the current 3D position of each object in the scene using Eq. 7, which selects the 3D position that minimizes the sum of the stereoscopic reconstruction errors computed for each view. To construct 3D object trajectories, we must to solve another data association problem, the assignment of current 3D object positions to the 3D tracks established in previous time steps. We can solve this problem indirectly by determining, for each of the N camera

views separately, the assignment of the 2D projections of current object positions to the 2D tracks established in previous time steps. For each object in each camera view, we use a 2D Kalman filter to predict the object position in the next frame. Across-frame data association can then be accomplished by matching each 2D object track to the 2D measurement that is closest to the predicted 2D object position.

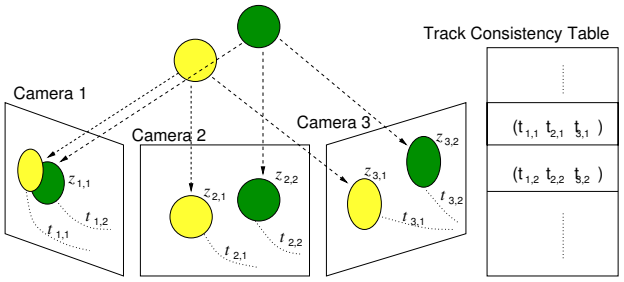

Figure 3. Example of the tracker-to-tracker fusion. Two objects are observed in three cameras, and there are two separate 2D trackers $f_{i,1}, f_{i,2}$ for each camera i. Based on the tracking history, the tracker-to-tracker associations are maintained in the track consistency table, e.g., tracker $f_{1,1}$ from camera 1, tracker $f_{2,1}$ from camera 2, tracker $f_{3,1}$ from camera 3 are associated to form an entry in the table. When occlusion occurs in camera 1, either $f_{1,1}$ or $f_{1,2}$ will lose track when they "compete" for measurement $z_{1,1}$. However, by looking at the solution of association across views $\{(z_{1,1}, z_{2,1}, z_{3,1}), (z_{1,1}, z_{2,2}, z_{3,2})\}$ and checking the entries in the table, the 2D tracker that initially lost the competition for $z_{1,1}$ in camera 1 can recover and "claim" measurement $z_{1,1}$. As a result, $z_{1,1}$ is associated with the tracks maintained by both trackers. By a similar mechanism, our system recovers from the track-switch problem.

The 2D across-time data association method will likely result in ambiguities and mismatches due to occlusions in densely populated scenes. If objects do not distinguish themselves by unique moving directions, the occlusions must be resolved to prevent track lost or track switch. We therefore analyze the across-view correspondences, established with IGRASP in each time step, which should be consistent through time. In particular, measurements of an object that are associated at time t should correspond to tracks that have been associated at time $t-1$. We maintain a consistency table during tracking that records the consistency of correspondence across views (Fig. 3). If some measurements $(z^{(t)}_{1,i_1}, z^{(t)}_{2,i_2}, ..., z^{(t)}_{N,i_N})$ are associated at current time step, their associated 2D trackers $(f_{1,j_1}, f_{2,j_2}, ..., f_{N,j_N})$ form an entry in the consistency table. Here the tracker f_{k,j_k} tracks measurement z_{k,i_k} in view k independently. If the 2D trackers perform well, this entry should be maintained in the table until some tracker ends. However, if some 2D tracker incorrectly associates a measurement in its own view, it will be corrected by looking at the corresponding entry in the table and historically comparing its consistency. The correction is performed only when the associations across at least $\lceil N/2 \rceil$ views are consistent (e.g., in 3-camera case, two consistent interpretations are needed). The consistency table also provides a good partial feasible solution for the assignment problem because measurements tracked by established trackers $(f_{1,j_1}, f_{2,j_2}, ..., f_{N,j_N})$ are very likely to be associated again. By comparing the assignments computed by IGRASP with the track tuples listed in the consistency table, our system can also prevent assignments that could mistakenly result in a switch of tracks.

OCCLUSION REASONING FOR 3 VIEWS IN ONE TIME STEP

Input: Current measurements $\{z_{s,i_s}\}$ and 2D-tracks $\{f_{s,i_s}\}$

I. Within each view s, assign measurements z_{s,i_s} to 2D-tracks f_{s,j_s} using bipartite matching.

II. Run IGRASP to find across-view associations of measurements $\{(z_{1,i_1}, z_{2,i_2}, z_{3,i_3})\}$ and construct the track-to-track associations $\{(f_{1,i_1}, f_{2,i_2}, f_{3,i_3})\}$.

III. Check if the track-to-track association tuples are consistent with entries in the Track Consistency Table (Fig. 3).

For each tuple $\mathbf{f} \in \{f_{1,i_1}, f_{2,i_2}, f_{3,i_3}\}$:

- **New Track:** If tuple $\mathbf{f}$ consists of at least two track labels that do not appear in the table, insert $\mathbf{f}$ into the table as a new entry.
- **Occlusion and Lost Track:** If tuple $\mathbf{f}$ is partially matched to some entry in the table (i.e., 2 of 3 track labels match), its label f_{s,i_s} differs from f_{s,i'_s} in the table entry, and 2D-track f_{s,i'_s} was found lost in step I, then assign measurement z_{s,i_s} to f_{s,i'_s}.

For remaining tuples, pair them and check:

- **Track Switch:** If two tuples are partially matched to table entries a and b (i.e., 2 of 3 track labels match), label $f_{s_1,i_{s_1}}$ in tuple 1 differs from the label in a, label $f_{s_2,i_{s_2}}$ in tuple 2 differs from the label in b, and a track label switch results in a match for both a and b, then reassign $z_{s_1,i_{s_1}}$ to $f_{s_2,i_{s_2}}$ and $z_{s_2,i_{s_2}}$ to $f_{s_1,i_{s_1}}$.

IV. Within each view, predict 2D-track state with Kalman filter based on assignments updated in step III.

The idea behind our method is essentially tracker-to-tracker sensor fusion. We maintain sensor-level trackers for each view and adjust their individual estimations after finding correspondences across views. Our distributed tracking style is extremely important if the communication overload or burden of a central computing node need to be minimized. The alternative is to collect all measurements from each view, reconstruct their 3D positions, and apply recursive Bayesian tracking in 3D space. We do not currently follow this centralized style because the reconstructed 3D positions are not sufficiently accurate due to sub-optimal across-view associations and detection errors. Future work

will compare the performance of the two approaches.

3. Experiments and Results

Observing the flight behavior of large groups of bats or birds is fascinating – their fast, collective movements provide some of the most impressive displays of nature. Quantitative studies of cooperative animal behavior have typically been limited to sparse groups of only a few individuals. The major limitation in these studies has been the lack of tools to obtain accurate 3D positions of individuals in dense formations. Although important progress has been made [3], robust general solutions to 3D tracking, reconstruction, and data association have been lacking. In our experiments, we first validated our method using synthetic data for which we had ground truth and then applied it to infrared thermal video of colonies of Brazilian free-tailed bats. We collected this video while the colony was emerging from its cave roost at night. We reconstructed the 3D flight paths and thus provided the first stereoscopic analysis of the emergence behavior of free-ranging bats.

3.1. Validation of Across-View Data Association

We generated synthetic data to test the performance of our IGRASP using a particle dynamics environment (Autodesk Maya). To simulate the scene near a cave in Texas where we recorded emerging bats, we generated spherical particles, 28 cm in radius, to move in a 20 x 5 x 5 m^3 space at a fixed speed of 2 m/s. We experimented with incrementally increasing emergence rates between 1 and 100 particles per second. Sample images with a high degree of density of particles are shown in Fig. 4. The trajectories were randomized by placing an axial and radial constraint on the particle movement. Three virtual cameras with overlapping views were positioned laterally and slightly below the average direction of travel of the particles. Since the calibration parameters for each camera and the 3D positions of each particle are known (i.e., the "ground truth"), we can test whether our solution of the multidimensional assignment problem (Eq. 9) matches particles correctly that are detected in the three views.

We demonstrate the performance of our IGRASP as a function of different particle densities in Fig. 5. As the number of particles increases, an increasing number of particles share overlapping regions in each field of view, which can then be detected as a single measurement. We measure as the "overlap density," the ratio of number of overlapping particle projections over the total number of particles (Fig. 5 left), and also the ratio of correct matches as number of correct tuples found by IGRASP over the ground truth (Fig. 5 right). Our results show that even in very dense scenarios, IGRASP can recover up to 65% matches correctly. When 20 particles/s are generated, 105 particles on average appear

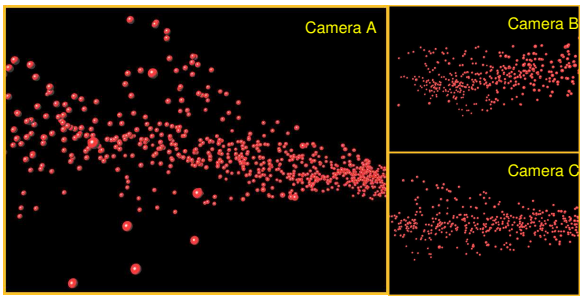

Figure 4. Data used for validation of across-view association. The three views show a scenario in which particles emerged at the right of the fields of view at a rate of 100 particles/s and moved towards the left side.

in a frame with an overlap density of 16%, and 95 % of the matches IGRASP computes are correct.

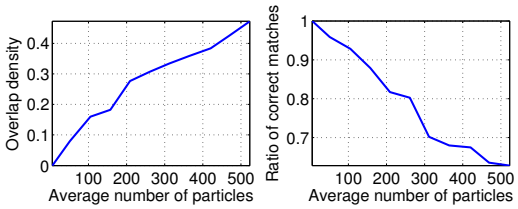

Figure 5. Across-view data association performance of IGRASP.

IGRASP has very few parameters to be adjusted. The execution time of the algorithm depends on the sparsity of the multidimensional assignment problem. We can use the epipolar constraint to reduce the number of feasible candidate tuples (Sec. 2.1). This turns out to be very important for the overall efficiency of the method. Computing the cost for all feasible tuples is much more expensive than determining the assignments (Fig. 6). We limit the costs of the time-consuming Building Phase using a critical threshold τ as follows: Only those measurements whose distances to the epipolar lines are within threshold τ are considered to form feasible tuples. A drawback of using a reduced feasible set is that IGRASP may not find the optimal set of assignments. Thus, parameter τ plays an important role in trading off accuracy and efficiency in dense tracking scenarios. The number of iterations $maxIter$ affects the optimality of IGRASP: when the number becomes large, IGRASP approaches exhaustive enumeration. We set $maxIter = 20$ throughout our experiments because its increase did not improve the performance significantly.

3.2. Infrared Thermal Video Analysis

We recorded the emergence of a colony of Brazilian free-tailed bats from a natural cave in Blanco County, Texas. We used three FLIR SC6000 thermal infrared cameras with a resolution of 640 $\times$ 512 pixels at a frame rate of 125 Hz (Fig. 7). We implemented our algorithms in C++ and tested our system on a Intel Pentium 2.36 GHz platform. Pro-

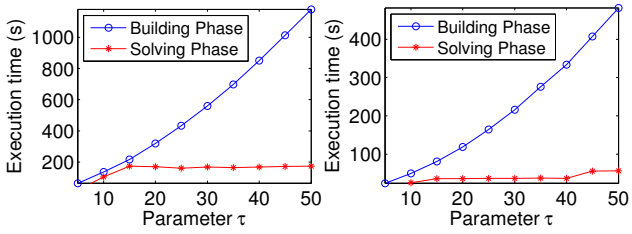

Figure 6. Execution time of IGRASP (our Matlab version) with different values of τ for the across-view data assignment. Left: 100 particles/s. Right: 50 particles/s.

cessing is performed in near real time and depends on the density of the group (e.g., in a 100 bats/frame scenario, our system took 3 s to process each frame).

Our experiments showed that we can track each individual bat in the emergence column, reconstruct their 3D flight paths, and provide insights into their group behavior based on trajectory analysis.

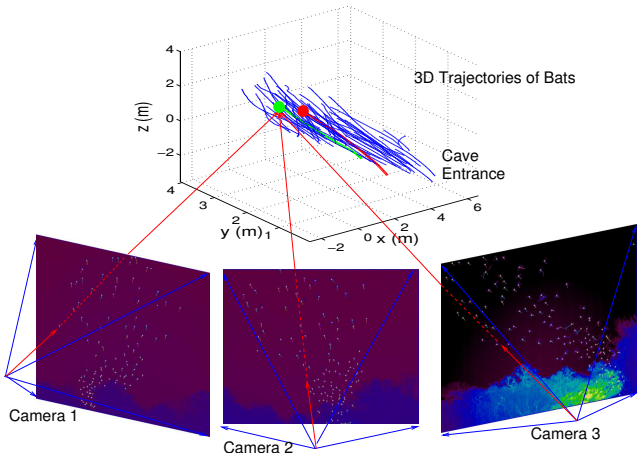

Figure 7. Visualization of camera setup and 150 reconstructed 3D trajectories. The camera baselines are ≈ 1 m. We adjusted camera pitch, yaw and roll to capture the full volume of the 3D column of emerging bats in overlapping field of views. In each view, there were as many as 200 bats at the same time, with an average size of 5×5 pixels. The average speed of an emerging bat was 8.75 m/s. The average direction of the emerging column can be described by the Euler angles 127°, 97° and 38°.

To detect moving bats, we applied adaptive background subtraction to identify the connected components of interest and then used the pixel with the highest intensity within each component as the position of a bat. We implemented 2D Kalman filters to track bats in each view and solved the across-time data association with bipartite matching. If a 2D tracker identifies a track loss, it keeps searching along the projected 2D flight path of the bat for the next 5 frames to see if it can resume tracking the bat or if it needs to wait for a reassignment of a measurement when across-view associations are solved and tracker-to-tracker consistency is checked.

Our results show that our method correctly resolves ambiguities due to occlusions (Fig. 8). However, we cannot expect to resolve all ambiguities in dense tracking situations due to insufficient image resolution. We investigated the performance of our system in resolving occlusions in scenarios with four different density levels of the column of emerging bats (Table 1). We counted the number of times each 2D tracker claimed to be lost for all three views. If the system could not resolve occlusion, it generated a new tracker once the bats were separated again. The number of computed tracks is therefore usually higher than the true number of bats. In relatively sparse scenarios, our system successfully recovered from occlusions and avoided track switches (40/56=74%). In the highly dense cases, occlusions typically occurred in two or three views at the same time, and so it was significant that we could correctly interpret 88/368=24% of the occlusions.

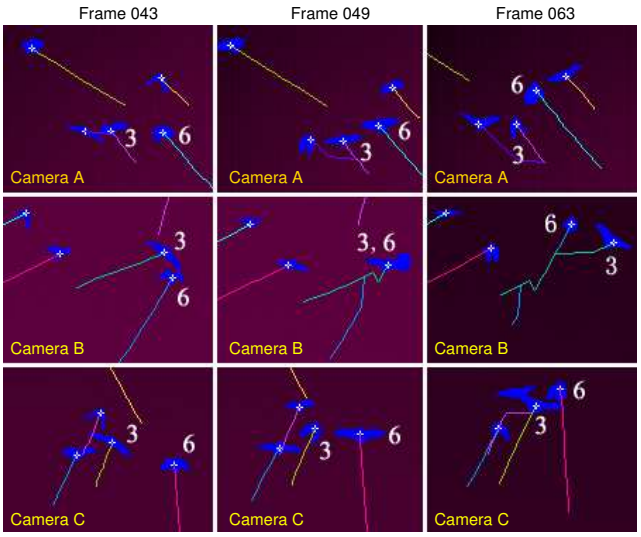

Figure 8. Occlusion Interpretation. Bats tracked in infrared video from multiple views are shown as segmented foreground objects (blue) with their tracker number (white). Frames 43, 49, and 63 are snapshots before, during, and after occlusion occurred in the field of view of Camera B, involving four bats that were flying close to each other. In particular, Bats 3 and 6 were difficult to distinguish in frame 49 recorded by camera B. Since their projections were well separated in the other two views during the period of occlusion, our algorithm was able to correctly interpret the occlusion by reasoning about the 3D positions of the four tracked bats. The output from the algorithm indicates that Bat 3 occluded Bat 6 in frame 49 recorded by camera B.

We reconstructed the full 3D trajectories of 150 bats and explored their group behavior during emergence. We measured the average emerging speed of a bat to be 8.75 m/s ($\approx$ 20 miles/h), which is consistent with the low end of the range of emergence speeds reported in the mammalogy literature. We have also resolved the question about the Euclidean distance between emerging bats. Our results

Table 1. Performance of tracking system in resolving occlusions. Ground truth was established by manual marking of four 100-frame sequences.

# of Bats/frame	True # of Bats	Computed # of Tracks	# of Occlusions	# of Recovered Occlusions
20	25	33	56	40
40	50	63	94	54
60	71	90	140	86
100	119	185	368	88

show that when 1–15 bats emerge, their average distance is 90 cm. The average distance drops to 35–45 cm as soon as the emergence column contains more than 25 bats per second (Fig. 9).

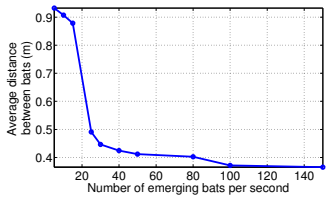

Figure 9. Results of 3D trajectory analysis: Average distance between emerging bats as a function of emerging rate, expressed by the average number of emerging bats per second.

4. Conclusions and Future Work

Our experiments showed that our method can reconstruct 3D trajectories of tightly-spaced, fast-moving objects and can accurately settle track uncertainties that could not be resolved from single views due to occlusion.

Our work can be extended to incorporate both across-time and across-view associations at the same time in a single optimization framework. It would be interesting to determine whether forward and backward inferences on the assignment over time could enhance the performance of our approach for highly dense groups. We also plan to do additional data mining on the group behavior of bats, once we generated hundreds of thousands of trajectories, which will be extremely valuable for scientists in other fields.

References

[1] S. Ali and M. Shah. Floor fields for tracking in high density crowd scenes. In *ECCV*, 2008.

[2] E. Andrade, S. Blunsden, and R. Fisher. Performance analysis of event detection models in crowded scenes. In *Workshop "Towards Robust Visual Surveillance Techniques and Systems" at VIE*, 2006.

[3] M. Ballerini, N. Cabibbo, R. Candelier, A. Cavagna, E. Cisbani, I. Giardina, V. Lecomte, A. Orlandi, G. Parisi, A. Procaccini, M. Viale, and V. Zdravkovic. Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. *Proceedings of the National Academy of Sciences*, 105:1232–1237, 2008.

[4] Y. Bar-Shalom and X. R. Li. *Multitarget - Multisensor Tracking: Principles and Techniques*. YBS Publishing, 1995.

[5] M. Betke, D. E. Hirsh, A. Bagchi, N. I. Hristov, N. C. Makris, and T. H. Kunz. Tracking large variable numbers of objects in clutter. In *CVPR*, 2007.

[6] G. J. Brostow and R. Cipolla. Unsupervised Bayesian detection of independent motion in crowds. In *CVPR*, 2006.

[7] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom. A generalized s-d assignment algorithm for multisensor-multitarget state estimation. *IEEE Trans. AES*, 33, 1997.

[8] R. Eshel and Y. Moses. Homography based multiple camera detection and tracking of people in a dense crowd. In *CVPR*, 2008.

[9] R. I. Hartley and A. Zisserman. *Multiview view geometry in computer vision*. Cambridge University Press, 2003.

[10] Y. Huang and I. Essa. Tracking multiple objects through occlusions. In *CVPR*, pages 1051–1058, 2005.

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

[12] Z. Khan, T. Balch, and F. Dellaert. MCMC data association and sparse factorization updating for real time multitarget tracking with merged and multiple measurements. *IEEE Trans. PAMI*, 28:1960 – 1972, December 2006.

[13] Y. Li, A. Hilton, and J. Illingworth. A relaxation algorithm for real-time multiple view 3d-tracking. *Image Vis Comput*, 20:841–859, 2002.

[14] A. Mittal and L. S. Davis. M2Tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *IJCV*, 51(3):189–203, 2003.

[15] K. Otsuka and N. Mukawa. Multiview occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *CVPR*, pages 90–97, 2004.

[16] A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, and W. Hu. Multi-object tracking through simultaneous long occlusions and split-merge conditions. In *CVPR*, 2006.

[17] V. Rabaud and S. Belongie. Counting crowded moving objects. In *CVPR*, 2006.

[18] A. J. Robertson. A set of greedy randomized adaptive local search procedure (GRASP) implementations for the multi-dimensional assignment problem. *Computational Optimization and Applications*, 19(2):145–164, 2001.

[19] A. Veeraraghavan, R. Chellappa, and M. Srinivasan. Shape-and-behavior-encoded tracking of bee dances. *IEEE Trans. PAMI*, 30:463 – 476, March 2008.

[20] X. Wang, K. Ma, G. Ng, and W. Grimson. Trajectory analysis and semantic region modeling using a nonparametric bayesian model. In *CVPR*, 2008.

[21] Q. Yu, G. Medioni, and I. Cohen. Multiple target tracking using spatio-temporal Markov Chain Monte Carlo data association. In *CVPR*, pages 1–8, 2007.

[22] T. Yu and Y. Wu. Collaborative tracking of multiple targets. In *CVPR*, pages 834–841, 2004.

[23] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.