Reconstructing 3D motion trajectories of particle swarms by global correspondence selection

Danping Zou[1], Qi Zhao[2], Hai Shan Wu[1], and Yan Qiu Chen[1]
[1]School of Computer Science
[2]School of Information Science and Engineering
Fudan University, Shanghai, China
{dpzou|zhaoqi|hswu|chenyq}@fudan.edu.cn

Abstract

This paper addresses the problem of reconstructing the 3D motion trajectories of particle swarms using two temporally synchronized and geometrically calibrated cameras. The 3D trajectory reconstruction problem involves two challenging tasks - stereo matching and temporal tracking. Existing methods separate the two and process them one at a time sequentially, and suffer from frequent irresolvable ambiguities in stereo matching and in tracking.

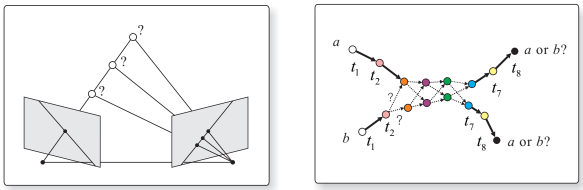
We unify the two tasks, and propose a Global Correspondence Selection scheme to solve stereo matching and temporal tracking simultaneously. It treats 3D trajectory acquisition problem as selecting appropriate stereo correspondences among all possible ones for each target by minimizing a cost function. Experiment results show that the proposed method has significant performance advantage over existing approaches.

1. Introduction

Phenomena that can be abstracted as particle swarms such as insect swarms, bird flocks, and fish schools occur prevalently in our environments. They have attracted significant attention by scientists in many disciplines [12, 10, 3]. The availability of the 3D motion trajectory for each individual in the swarm can greatly facilitate the study of their collective behavior. There however has been little research towards developing effective methods to achieve this purpose.

A feasible way to measure the 3D motion trajectories is through using multiple video cameras. The cameras capture these dynamic targets from different viewing directions, and record their positions projected onto the 2D image planes at each time step. The problem we need to solve is to retrieve the time-varying 3D locations of these targets from the video sequences. It involves two challenging tasks, namely, stereo matching - establishing the stereo correspondences across views - and tracking - finding motion correspondences for the particles. There have been several approaches proposed to accomplish such tasks. These approaches can be classified into two main groups.

In the first category, stereo correspondences are first established at each frame to reconstruct 3D locations of the targets. The locations corresponding to the same target at different frames are temporally associated to yield the final 3D trajectory [8, 11, 13]. However, stereo matching ambiguities frequently arise, especially when binocular system is employed as shown in Figure 1(a) . Consequently, the final result would be greatly compromised by incorrect 3D locations caused by wrong matches.



(a) The point on the left image has multiple candidates on the right image satisfying epipolar constraint.

(b) During $t_3 - t_6$, the projections of two targets on the image plane become very close, making tracking difficult.

Figure 1. Ambiguities in stereo matching and 2D tracking.

In the second category [4, 7, 5], targets are first tracked throughout the image sequence captured by each camera. The resultant 2D tracks are then matched using camera geometry to generate 3D trajectories. For such methods, motion cue is used to disambiguate stereo matching. However, tracking of large quantity of targets in image sequence is frequently interfered by occlusion (which occurs when image projections of several targets overlap) and interaction (which occurs when multiple targets move closely) as shown in Figure 1(b).

In both strategies mentioned above, tracking and stereo

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

matching are artificially regarded as two independent successive stages. In fact, these two stages can facilitate each other. Motivated by this observation, we propose an unified method to disambiguate tracking and stereo matching simultaneously. We treat the reconstruction of 3D trajectories as selecting a sequence of stereo correspondences between the image projections of the targets throughout the entire time span. All the frames are taken into account, and the optimal trajectories are obtain by minimizing a cost function incorporating three cues, namely epipolar constraint, motion coherence, and configuration observation match.

Experimental results on synthetic data demonstrate that the proposed method exhibits significant advantage over existing approaches in presence of occlusion and severe interaction. We also test the proposed method on a challenging real-world case - a few hundreds of fruit flies flying freely in an acrylic box - and successfully obtain their 3D trajectories as the supplementary video shows.

The major contribution of this paper is twofold:

- We present a global optimization framework that incorporates all available cues to solve tracking and stereo matching simultaneously for large swarms of indistinguishable particle-like objects.

- We are the first to obtain complete 3D motion trajectories of a swarm of hundreds of flying fruit flies, enabling biologists to conduct thorough study of collective behavior of fruit flies.

2. Related work

Numerous approaches have been developed for multi-target tracking in image sequence, including MHT[14], JPDAF[1], Particle Filter[9], Greedy Assignment[17], and Track Graph[15]. Among them, Greedy Assignment(GOA) is regarded as the most efficient for dealing with large number of targets, and it has been used to track a large swarm of flying bats [2]. In [18], GOA was adopted to extract 2D tracks of flying bees from image sequence. By virtue of narrow baseline between cameras, the curvature information of these tracks was utilized to enhance the performance of stereo matching. However, all the above methods suffer from inherent ambiguity of 2D tracking. Likewise, approaches through stereo matching followed by tracking would also fail in presence of stereo matching ambiguities. A few methods incorporate tracking and stereo matching to generate more reliable results. Du et al. [4] segmented 2D tracks when interaction occurs, and then established correspondences among these segments. However, the common time span between segments could be too short to resolve stereo matching ambiguity. In addition, the resultant 3D trajectories are undesirably broken into many segments. Willneff et al. [19] proposed a spatio-temporal matching algorithm using motion prediction to eliminate stereo matching ambiguities. Unfortunately, the algorithm would fail in presence of stereo matching ambiguities, because generating reliable prediction highly depends on correct 3D locations in the previous frame. In addition, only several consecutive frames are taken into account, whereas our method processes the entire time span in a global manner.

3. 3D trajectory reconstruction

Consider a 3D swarm of flying targets of similar appearance and small size. They are recorded by two geometrically calibrated and temporally synchronized cameras from different viewing directions. At regular time steps $t = 1, 2, \ldots, T$, the cameras capture these targets, producing image blobs with centers being $M_t = \{m_t^{v,i}\}$, where $v \in \{1, 2\}$ indicates the v-th view and $i \in \{1, 2, \ldots, N_t^v\}$ labels the i-th blob in the v-th view.

To recover the 3D trajectories of the targets, we need to establish stereo correspondence between blobs of different views over time. We regard each pairing of blobs as a potential stereo correspondence. The set of all possible pairings at time step t is given by $H_t = \{m_t^{1,i}\} \times \{m_t^{2,j}\}$. A pairing could be either true or false correspondence. Our goal is to correctly assign the true pairings to each target. Let $S_t = (s_t^1, s_t^2, \ldots, s_t^N)$ be a configuration of pairings assigned to the targets at time step t, where s_t^n represents the pairing chosen for target n. Our mission is then to find a sequence of configurations $S_{1:T} = (S_1, \ldots, S_T)$ that best explain the image blobs recorded during the capturing process. Thereafter, the 3D motion trajectory of each target can be reconstructed from respective sequence of stereo correspondences $s_{1:T}^n = (s_1^n, \ldots, s_T^n)$ through triangulation.

4. Global correspondence selection

We propose a Global Correspondence Selection (GCS) scheme to solve the problem via finding an optimum configuration sequence $S_{1:T}^*$ that minimizes a cost function $f(S_{1:T})$, that is:

$$S_{1:T}^* = \arg\min_{S_{1:T}} f(S_{1:T}). \tag{1}$$

The cost function $f(\cdot)$ is designed to incorporate three cues - epipolar constraint, kinetic coherency and configuration projection match. The details are discussed in following sections.

4.1. Epipolar constraint

If the blobs in different views correspond to the same 3D target, they should satisfy the epipolar constraint. This is an important cue for judging how likely a pairing of blobs is a true stereo correspondence. Given a paring, the cost of

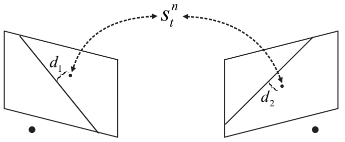


Figure 2. The average distance is defined as $\rho_e(s_t^n) = (d_1 + d_2)/2$.

being corresponding to the same target is defined as

$$f_e(s_t^n) = \rho_e(s_t^n) \tag{2}$$

where $\rho_e(\cdot)$ represents the average distance between the blob centroids and their respective epipolar lines as shown in Figure 2. To rule out the apparently false pairings whose blobs are far from respective epipolar lines, we set $f_e(s_t^n) = \infty$, if $\rho_e(s_t^n)$ is greater than a threshold ϵ_e. For all pairings selected at time t, the total cost of their being true correspondences is obtained by

$$f_E(S_t) = \sum_{n=1}^{N} f_e(s_t^n). \tag{3}$$

4.2. Kinetic coherency

Each pairing gives rise to a 3D location through triangulation. A genuine pairing sequence should generate 3D locations that is feasible as a trajectory traveled by a physical object. To evaluate the likelihood of being a trajectory, kinetic model is used. At each time instance, the kinetic model gives a prediction of the motion state of the target based on previous states. A trajectory is considered more likely to be traveled by a target if the overall prediction error is smaller.

This is an important cue for selecting appropriate pairing sequence for a target. Consider the sequence of pairings $s_{1:T}^n = (s_1^n, \dots, s_T^n)$ assigned to target n and their corresponding 3D locations denoted by $\mathbf{x}_1, \dots, \mathbf{x}_T$. The prediction inaccuracy at time t is defined as

$$f_k(s_{t-K}^n, \dots, s_t^n) = \rho_k(\tilde{\mathbf{x}}_t, \mathbf{x}_t). \tag{4}$$

The predicted 3D location $\tilde{\mathbf{x}}_t$ is computed from K previous states using the kinetic model, namely $\tilde{\mathbf{x}}_t = \mathscr{P}(\mathbf{x}_{t-K}, \dots, \mathbf{x}_{t-1})$. The function $\rho_k(\cdot, \cdot)$ is the Euclidean distance between two 3D locations.

Selection of kinetic model relies on priori knowledge of the motion of the subjects. We adopt here a simple and yet powerful kinetic model - nearest-neighbor model [17], i.e., $K = 1$ and $\mathscr{P}(\mathbf{x}_{t-1}) = \mathbf{x}_{t-1}$. Although nearest-neighbor model does not accurately describe the motion in most cases, it has been proved to be an effective model adopted in many tracking applications, particularly when the motion is complex and hard to be formulated, such as wandering people and drifting insects. To reduce the number of candidate trajectories, we assign a threshold ϵ_k to remove impossible candidates by setting $f_k(s_{t-1}^n, s_t^n) = \infty$ when $\rho_k(\mathbf{x}_{t-1}, \mathbf{x}_t) > \epsilon_k$. By summing up the prediction errors of all targets, we get the total kinetic cost of successive pairings:

$$f_K(S_{t-1}, S_t) = \sum_{n=1}^{N} f_k(s_{t-1}^n, s_t^n). \tag{5}$$

4.3. Configuration observation match

Another cue comes from measuring the level of match between configuration and observation. We assume that: 1) cameras are well placed so that they can capture most targets simultaneously; 2) at each time step, the proportion of overlapping image blobs on the 2D image plane is relatively small. The first assumption holds in most cases; the second one is also reasonable when the number of flying targets is moderate - hundreds to thousands. The two assumptions statistically imply that, in each view, one blob corresponds to only one particular target and vice versa.

Having noticed the tendency of one-to-one mapping between blobs and targets, we propose a cost to evaluate how well a given configuration S_t accounts for the observed blobs M_t:

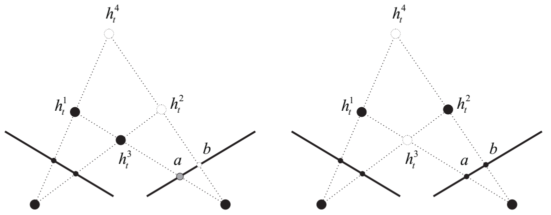
$$\begin{aligned} f_C(S_t, M_t) =& \frac{1}{N_t^1} \sum_{i=1}^{N_t^1} |n_c(m_t^{1,i}, S_t) - 1| \\ &+ \frac{1}{N_t^2} \sum_{i=1}^{N_t^2} |n_c(m_t^{2,i}, S_t) - 1|, \end{aligned} \tag{6}$$

where $m_t^{v,i}$, N_t^v denote the blob centers and the total number in v-th view, and $n_c(\cdot, \cdot)$ represents the number of targets to which the blob is mapped with regard to current configuration of assignments. We use a threshold ϵ_c to prevent the configurations from mapping a blob to too many targets. That is, if $n_c(\cdot, \cdot) > \epsilon_c$, the match cost will be set to infinity, $f_c(S_t, M_t) = \infty$. Figure 3 shows the costs of different configurations.

5. Cost function and optimization method

Combining the above-discussed three terms additively, we obtain the overall cost function:

$$\begin{aligned} &f(S_{1:T}) = \\ &\alpha \sum_{t=1}^{T} f_E(S_t) + \beta \sum_{t=1}^{T} f_C(S_t, M_t) + \gamma \sum_{t=2}^{T} f_K(S_{t-1}, S_t), \end{aligned} \tag{7}$$



(a) $S_t = (h_t^1, h_t^3)$ (b) $S_t = (h_t^1, h_t^2)$

Figure 3. h_t^1, h_t^2, h_t^3, and h_t^4 are four possible pairings. The configuration in (b) is more desirable than that in (a). In (a), blob a relates to two targets but b has no target related; while in (b), one-to-one mapping between the blobs and targets are established in each view.

where α, β, γ are the weights of the terms. The cost function can be recursively decomposed into

$$f(S_{1:t}) = f(S_{1:t-1}) + \Delta f(S_t). \qquad (8)$$

where the increment of cost $\Delta f(S_t)$ is

$$\Delta f(S_t) = \alpha f_E(S_t) + \beta f_C(S_t, M_t) + \gamma f_K(S_{t-1}, S_t). \quad (9)$$

The cost function can be optimized by dynamic programming. It is accomplished in two stages: 1) proposing possible configurations for next frame, 2) propagating cumulative minimum cost from previous ones to each newly proposed configuration.

Since the number of possible configurations increases exponentially with the number of target, we use Gibbs sampling [6] to only obtain those configurations with low costs. The initial configurations at the first frame can be sampled from [1]

$$S_1 \sim P\left[f(S_1)\right], \qquad (10)$$

where $P(\cdot)$ denotes a probability function that is decreasing at positive interval such that configurations with low costs could be sampled with high probabilities. Here, we let $P(u) \propto \exp(-u)$. For each sample at t-th frame, $S_t^{(i)}$, the configurations of the next frame can be proposed by sampling

$$S_{t+1} \sim P\left[\Delta f(S_{t+1})\right]. \quad (S_t = S_t^{(i)}) \qquad (11)$$

The cumulative minimum cost of a newly sampled configuration $S_{t+1}^{(j)}$, denoted by $f^*(S_{t+1}^{(j)})$, can be computed from

$$f^*(S_{t+1}^{(j)}) = \min_i \left[f^*(S_t^{(i)}) + \Delta f(S_{t+1}^{(j)})\right]. \qquad (12)$$

The number of samples at each frame is prevented from being too large by removing those samples with large costs. Otherwise it will increase exponentially with the number of frames processed. Configuration sampling and cost propagation are performed frame by frame and finally produce the optimum result as shown in Figure 4. We can see that through optimization, both ambiguities in stereo matching and tracking can be resolved as shown in Figure 5.

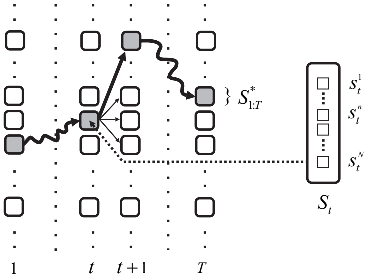


Figure 4. Optimization is done by dynamic programming - sampling new configurations at next frame and propagating cumulative minimum costs to them until reaching the last frame. Finally the optimum configurations sequence $S_{1:T}^*$ is obtained by tracing back.

6. Variable number of targets

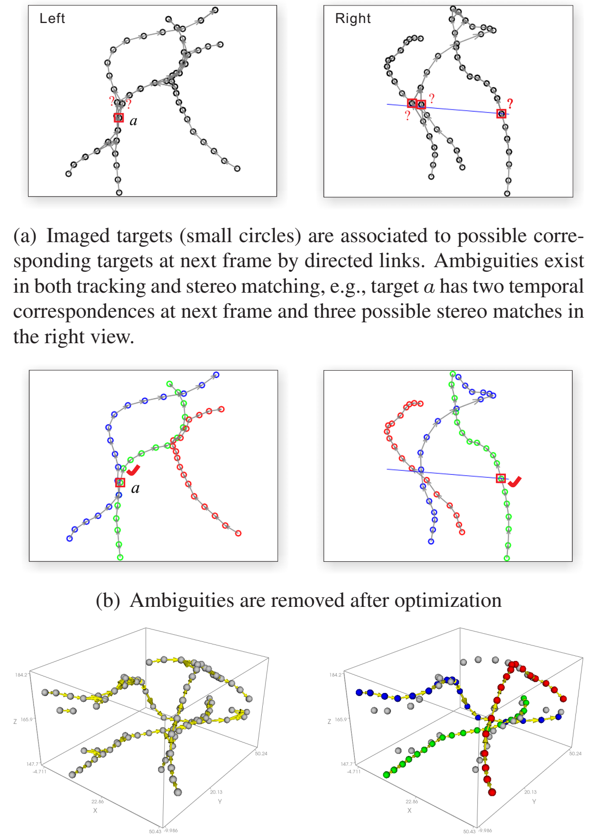
We have so far considered the case where the number of target is constant. The number of targets in practice, however, is variable since targets might enter and leave the scene. We introduce an additional variable $\mathcal{O}$ to represent the absent state of target. By setting the dimension of configuration large enough, the way to handle variable number of targets is similar to that of handling fixed one. The difference is that, at each frame, a dummy pairing $\mathcal{O}$ can be assigned to a target, indicating this target is not in the scene.

By introducing an additional pairing $\mathcal{O}$, the two costs of epipolar constraint and kinetic coherency needs to be modified. We redefine the two costs based on two rules: 1) the targets are encouraged to keep absent from the scene with zero costs; 2) a target tends to switch the visibility state only when it is impossible to find two pairings at adjacent video frames that can be assigned to this target with finite cost of kinetic coherence. The first rule indicates $f_e(\mathcal{O}) = 0$ and $f_k(\mathcal{O}, \mathcal{O}) = 0$, which ensures that whether a target is absent or not mainly relies on configuration observation match. The second rule suggests that the cost of visibility switching $f_k(h, \mathcal{O})$ (or $f_k(\mathcal{O}, h)$), denoted by η, should be properly set so that if there is a pairing h' that $f_k(h, h') < \infty$, we have $\beta f_k(h,h) + \alpha f_e(h) < \beta f_k(h, \mathcal{O}) + \alpha f_e(\mathcal{O})$. It yields $\eta > \alpha/\beta \cdot \epsilon_e + \epsilon_k$.

7. Efficient implementation

The above-disccussed optimization problem can be better understood by using a graph representation. We construct a directed graph as shown in Figure 5(c) where a node

[1] At this initial stage, samples with same assignments but in different orders, e.g. $S_1^{(1)} = (h_1, h_1')$ and $S_1^{(2)} = (h_1', h_1)$, should be treated as the same sample.



(a) Imaged targets (small circles) are associated to possible corresponding targets at next frame by directed links. Ambiguities exist in both tracking and stereo matching, e.g., target a has two temporal correspondences at next frame and three possible stereo matches in the right view.

(b) Ambiguities are removed after optimization

(c) Graph representation of pairings. Two pairings are connected by an edge weighted by the cost of kinetic coherency in (4).

(d) Pairing paths are extracted for each of the three targets (marked by red, greed and blue respectively) through optimization.

Figure 5. Through optimization of the cost function, ambiguities in both stereo matching and tracking are eliminated.

is the pairing satisfying the epipolar constraint; an edge connects two nodes and carries the cost of kinetic coherence at successive frames. Acquiring 3D trajectories is equivalent to finding N optimal paths in this graph. The number of target, N, can be set with regard to the maximum number of detected blobs in video streams. If N is small, the optimal paths can be obtained by direct optimization. But when the target number is large, direct optimization is infeasible, because the computational complexity is still too high in spite of employing sampling techniques. So we propose a two-phase strategy to significantly reduce the computational costs.

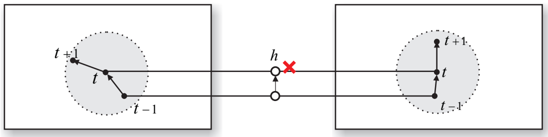


Figure 6. A false pairing h is detected, when both blobs leave respective epipolar lines of each other at $t+1$ without tracking ambiguity - no other blob moves into the adjacent regions indicated by gray disks from time $t-1$ to $t+1$.

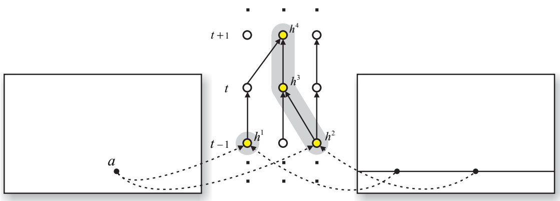


Figure 7. If a paring is temporally connected by more than one parings at adjacent frames - e.g., h^2, h^3, h^4 - or shares blobs with other parings - e.g., h^1 and h^2 share the blob a, it is an ambiguous pairing. The ambiguous pairings (yellow dots) are grouped into a cluster (indicated by gray shadow), if they are temporally connected or share blobs.

7.1. Graph simplification

The computational costs could be significantly reduced if many of false pairings included in the graph are identified and removed. A hypothetical target computed from false pairings would disappear abruptly when the paired blobs depart from their respective epipolar lines at sometime. This indicates that nodes with no out link in the graph are likely to be false pairings. To determine whether a node with no out link is a false pairing, we investigate motions of related blobs on the image planes. As shown in Figure 6, if both blobs move from previous frame to next frame without tracking ambiguity, the paring is identified as a false stereo correspondence. In the same manner, the false pairings which appear suddenly can also be recognized. The recognized false parings would be removed from the graph and the process is repeated until no more false pairing can be found.

7.2. Optimization in clusters

If no stereo-matching ambiguity exists in a pairing - a blob has only one corresponding blob on the related epipolar line in the other view - the two blobs can be established as a stereo correspondence. Through graph simplification, this kind of pairings emerge in large number. Many of them connect each other one-by-one and form a paring path without branches. These paths in fact produce most part of the result - 3D trajectory segments of targets. Therefore it is unnecessary to take them into account in optimization. We only focus on the remaining ambiguous pairings, which have either stereo-matching ambiguities or tracking ambiguities.

These ambiguous pairings are grouped into different clusters by introducing an equivalence relation, which is defined such that the two ambiguous pairings connected or sharing blobs are classified into the same cluster as shown in Figure 7.

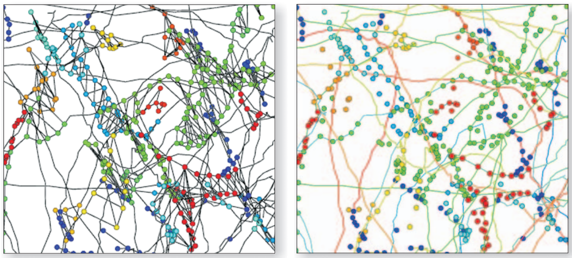


Figure 8. The ambiguous pairings(colored dots) are classified into clusters (marked by different colors). Optimization is done in each cluster and finally optimum paths are acquired (colored curves).

Optimization is done separately in each cluster. Each cluster is first extended by adding its adjacent pairings. Let $M_{\mathcal{A}^i}$ be the related blobs of the pairings in the cluster $\mathcal{A}^i$. We minimize the cost $f(S_{\mathcal{A}^i})$ to yield the optimum paths $S^*_{\mathcal{A}^i}$ for each cluster. The optimum paths of all clusters and the already obtained path segments are fused together, and finally produce the global optimum paths throughout all frames as shown in Figure 8. The technique significantly reduces the computational cost, enabling our method handle massive targets more efficiently.

8. Experiments

We test the proposed method on simulated particle swarms of various challenging levels. We also apply the proposed method to a real-world case of reconstructing the 3D trajectories of fruit fly swarms.

8.1. Simulated particle swarm

Simulated particles, confined in a cube of side length 2, are initialized with random locations and velocities. In each step, particle velocity is updated using

$$\mathbf{v}_t = \theta \mathbf{v}_{t-1} + \mathbf{n}_t \tag{13}$$

where $\theta \in [0, 1]$ is a parameter used to control the smoothness of velocity and $\mathbf{n}_t \sim \mathcal{N}(0, 0.05\mathbf{Id})$ ($\mathbf{Id}$ is a 3×3 identity matrix) is a Gaussian noise to perturb the velocity vector. Particle location is then computed from previous location by $\mathbf{x}_t = \mathbf{x}_{t-1} + \mathbf{v}_t$. At each time step, the particles are projected onto two image planes of 800×800 resolution from different views, simulating two cameras. Each particle is rendered as a sphere of the same color using OpenGL. The radius of the sphere is set to 0.02 to generate the projected balls in image sequences with diameter of 5 pixels. The random noises are added to simulate real imaging process.

Two cases are simulated to test the performance of the proposed method. In the first case, the average velocity is set around 5 pixels per frame, the number of targets varies from 10 to 100. In the second one, we change the average velocity from 6 pixels to 24 pixels per frame while the number of targets is set to 40. In both cases, θ is randomly chosen for each target around 0.9 to simulate the motion of flying insects. Evaluation metrics and results are presented in the next sections.

8.2. Evaluation metrics

To evaluate the results, we propose a metric named Reconstruction Association Error(RAE) which measures errors in both reconstruction and temporal association. It is defined as

$$RAE = \left[\frac{N_{mc} + N_{fc} + N_{ma} + N_{fa}}{2N \cdot T} \right], \tag{14}$$

where N and T are the numbers of targets and frames respectively. N_{mc} denotes the numbers of missing genuine correspondences. N_{fc} is the number of falsely selected correspondences. N_{ma}, N_{fa} are the numbers of missing associations and false associations between successive frames respectively.

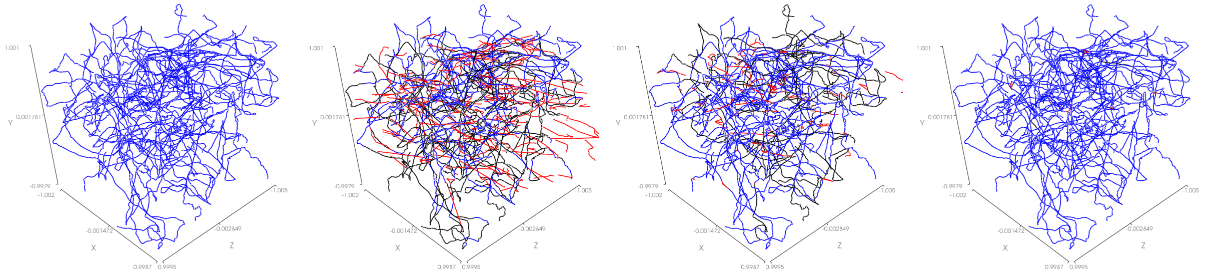
8.3. Results on simulated particle swarms

We compare our method with recent Relative Epipolar Motion (REM) method [4] that uses 2D trajectory-matching strategy, and an extension of REM adopting the GOA tracker [17] instead of nearest neighbor tracker. The reconstruction error and resulted 3D trajectories are shown in Figure 9 and Figure 10 respectively. In the first case, the increasing number of targets will result in more stereo matching ambiguities, producing more 2D tracking errors. Since GOA-REM utilizes motion prior to facilitate the 2D tracking, it outperforms REM. In the second case, due to the increasing tracking ambiguities, both methods fail to obtain 2D tracks efficiently. Unlike the above two methods, our approach achieves superior results in both cases, because it unifies tracking and stereo matching by global optimization in both spatial and temporal domain. As shown in Figure 11, the proposed method is resiliant to tracking ambiguity on 2D image plane.

8.4. Results on fruit fly swarm

The collective behavior of fruit flies has attracted significant attention from biologists[16, 10]. One important way is analyzing their 3D motion trajectories. We applied our approach to acquiring the 3D trajectories of large number of freely flying fruit flies.

The fruit flies flied in an acrylic glass box of size $35cm \times 35cm \times 25cm$, where the background was illuminated by white plane lights. Two synchronized and calibrated Sony HVR-V1C video cameras working in high speed mode of 200fps were used to capture the scene from different views.



(a) Ground truth (b) REM (RAE : 0.664) (c) GOA(RAE : 0.345) (d) The proposed GCS(RAE : 0.008)

Figure 10. Results of 60 targets using different methods. The black lines denote the missing correspondences/associations, and the red lines denote the false correspondences/associations. Our method yields remarkable result as in (C).

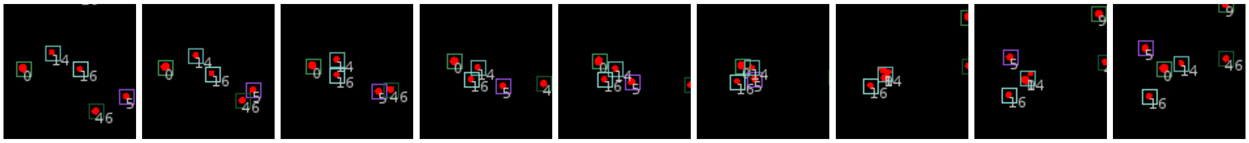


Figure 11. GCS yields correct tracking result, in spite of tracking ambiguity on 2D image planes.

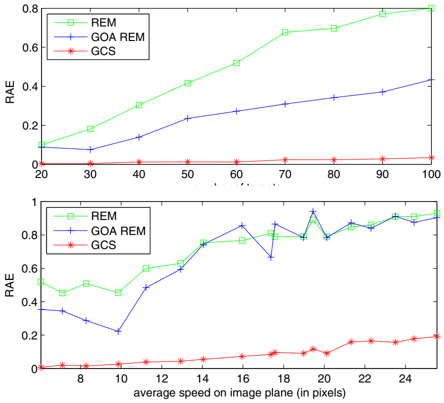


Figure 9. The proposed GCS method yields remarkable results in spite of high densities and fast speeds, while results from track-matching methods are poor, as errors reach up to 0.9.

The image resolution is 960 × 540. We first detected the fruit flies by background subtraction. We then applied our method to reconstruct the 3D trajectories of a sequence with 200 frames. We finally obtained 438 trajectories in total. The average length of these trajectories is 60.7 frames. Among all of them, 86 trajectories are longer than 100 frames. Figure 12 demonstrates the acquired 3D trajectories. At each frame, the blobs that correspond to reconstructed targets, in average reach 77.6% percent of detected blobs in each view. The result is promising, since some targets are not simultaneously captured by both cameras, particularly the targets near the image boundaries as shown in Figure 12.

9. Conclusion

We have proposed a novel approach for acquiring the 3D trajectories of a swarm of flying targets. The proposed method simultaneously solves tracking and stereo matching in a global manner, which significantly reduced ambiguities. The framework can further be extended in many ways: it can accommodate additional information such as texture and color if available; the pairing can be replaced by grouping in multiple cameras; high-order kinetic coherency can be adopted to achieve better results.

Acknowledgements

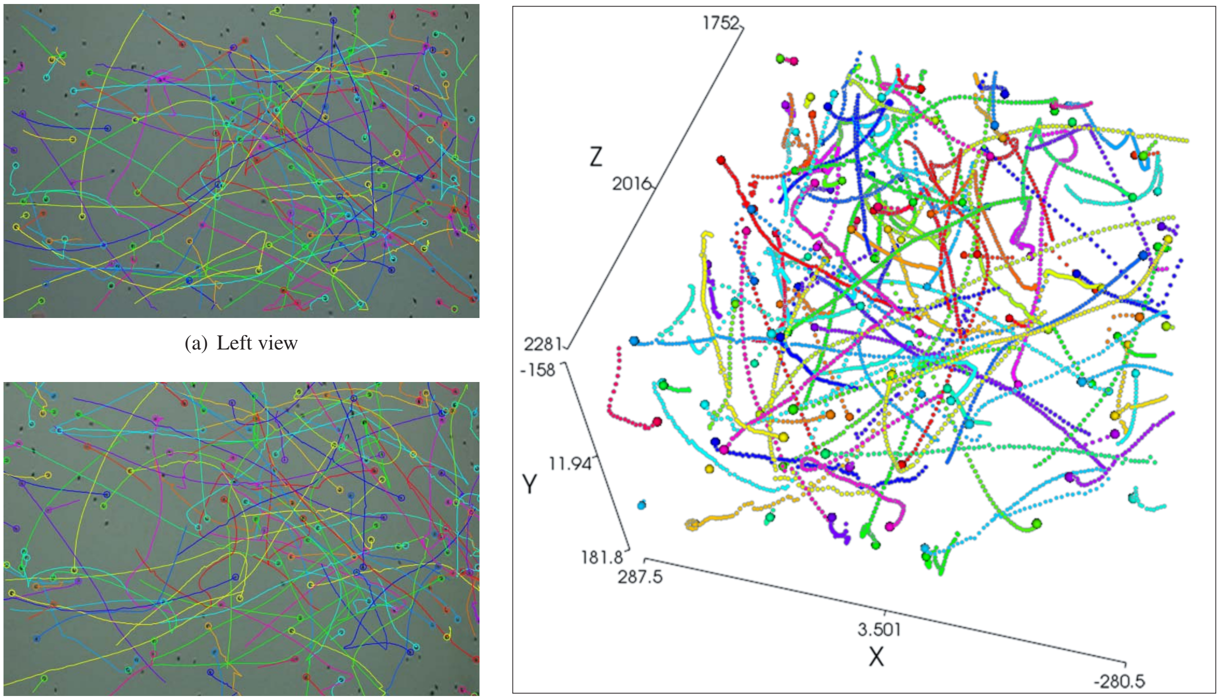
The research work presented in this paper is supported by National Natural Science Foundation of China, Grant No. 60875024. We would like to thank Linguo Li and Wei Li at School of Life Sciences, Fudan University for providing fruit flies for the experiments.

References

[1] Y. Bar-Shalom, T. Fortmann, and M. Scheffe. Joint probabilistic data association for multiple targets in clutter. In *Proc. Conf. on Information Sciences and Systems*, pages 404–409, 1980.

[2] M. Betke, D. Hirsh, A. Bagchi, N. Hristov, N. Makris, and T. Kunz. Tracking large variable numbers of objects in clutter. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[3] M. Betke, D. Hirsh, N. Makris, G. McCracken, M. Procopio, N. Hristov, S. Tang, A. Bagchi, J. Reichard, J. Horn, et al. Thermal imaging reveals significantly smaller brazilian free-tailed bat colonies than previously estimated. *Journal of Mammalogy*, 89(1):18–24, 2008.



(a) Left view

(b) Right view

(c) Reconstructed 3D trajectories

Figure 12. Results of experiment on fruit fly swarm. See more details in the supplemental video.

[4] H. Du, D. Zou, and Y. Chen. Relative epipolar motion of tracked features for correspondence in binocular stereo. In *IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007.

[5] D. Engelmann, C. Garbe, and M. Stöhr. Stereo particle tracking. In *Proc. of 8th Int. Symp. on Flow Visualization (CD-ROM)*, pages 240–1, 1998.

[6] S. GEMAN and D. GEMAN. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[7] Y. Guezennec, R. Brodkey, N. Trigui, and J. Kent. Algorithms for fully automated three-dimensional particle tracking velocimetry. *Experiments in Fluids*, 17(4):209–219, 1994.

[8] N. KASAGI and K. NISHINO. Probing turbulence with three-dimensional particle-tracking velocimetry. *Experimental thermal and fluid science*, 4(5):601–612, 1991.

[9] J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. *International Journal of Computer Vision*, 39(1):57–71, 2000.

[10] G. Maimon, A. D. Straw, and M. H. Dickinson. A simple vision-based algorithm for decision making in flying drosophila. *Current Biology*, 18(6):464–470, March 2008.

[11] N. Malik, T. Dracos, and D. Papantoniou. Particle tracking velocimetry in three-dimensional flows - part ii:particle tracking. *Experiments in Fluids*, 15(4):279–294, 1993.

[12] K. Norris and C. Schilt. Cooperative societies in three-dimensional space: on the origins of aggregations, flocks, and schools, with special reference to dolphins and fish. *Ethology and Sociobiology*, 9(2-4):149–179, 1988.

[13] F. Pereira, H. Stuer, E. Graff, and M. Gharib. Two-frame 3d particle tracking. *Measurement Science and Technology*, 17(7):1680–1692, 2006.

[14] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, 1979.

[15] J. Sullivan, S. Carlsson, and E. Hayman. Tracking and labelling of interacting multiple targets. pages 619–632, 2006.

[16] L. Tammero and M. Dickinson. The influence of visual landscape on the free flight behavior of the fruit fly drosophila melanogaster. *Journal of Experimental Biology*, 205(3):327–343, 2002.

[17] C. Veenman, M. Reinders, and E. Backer. Resolving motion correspondence for densely moving points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 54–72, 2001.

[18] A. Veeraraghavan, M. Srinivasan, R. Chellappa, E. Baird, and R. Lamont. Motion based correspondence for 3d tracking of multiple dim objects. In *IEEE Conference on Acoustics, Speech and Signal Processing*, volume 2, 2006.

[19] J. Willneff and A. Gruen. A new spatio-temporal matching algorithm for 3d-particle tracking velocimetry. In *The 9th International Symposium on Transport Phenomena and Dynamics of Rotating Machinery, Honolulu, Hawaii, USA, February*, pages 10–14, 2002.