Trajectory association and fusion across partially overlapping cameras

Nadeem Anjum and Andrea Cavallaro [*]
Queen Mary University of London
Multimedia and Vision Group
Mile End Road, E1 4NS London (United Kingdom)
{nadeem.anjum, andrea.cavallaro}@elec.qmul.ac.uk

Abstract

We present a novel unsupervised inter-camera trajectory correspondence algorithm that does not require prior knowledge of the camera placement. The approach consists of three steps, namely association, fusion and linkage. For association, local trajectory pairs corresponding to the same physical object are estimated using multiple spatio-temporal features on a common ground-plane. To disambiguate spurious associations, we employ a hybrid approach that utilizes the matching results on the image- and ground-plane. The trajectory segments after association are fused by adaptive averaging. Finally, linkage integrates segments and generates a single trajectory of an object across the entire observed area. We evaluated the performance of the proposed approach on a simulated and two real scenarios with simultaneous moving objects observed by multiple cameras and compared it with state-of-the-art algorithms. Convincing results are observed in favor of the proposed approach.

1. Introduction

The reconstruction of objects' trajectories across cameras facilitates the recognition of global behaviors for large scale events in applications such as sports analysis, remote sensing and video surveillance. This requires a mechanism for associating and integrating partially observed data in each camera view. Local trajectory information from individual cameras may be corrupted by inaccuracies due to noise, objects re-entrances, occlusions and by errors due to crowded scenes. Therefore, trajectory association becomes a difficult task under such complex scenarios.

In this paper, we consider the problem of object association across partially overlapping cameras using local trajectories. Existing works perform association either on image-plane [7] or on ground-plane [4]. As image-plane trajectories are heavily affected by the perspective deformations, which cause inaccurate associations especially if the trajectories are far from cameras. On the other hand, accurate associations on ground-plane are hampered by the image- to ground-plane projections, which do not ensure unique association of an object trajectories observed in multiple cameras. We propose a hybrid approach that combines the strength of both image- and ground-plane associations. Initial correspondence among trajectories is established on ground-plane using multiple spatio-temporal features and then image-plane reprojections of the matched trajectories are employed to resolve conflicting situations. This makes sure that only one trajectory of an object from each camera is associated to other cameras. The fusion is then applied to combine matched trajectories. A spatio-temporal linkage procedure connects the fused segments in order to obtain the complete global trajectories across the distributed setup. Figure 1 shows the proposed flow diagram.

The rest of the paper is organized as follows: Sec. 2 covers prior works in the field of object correspondence across multiple cameras. Section 3 provides the detailed description of the global ground-plane trajectories construction from local image-plane segments. Section 4 covers the experimental results and finally Sec. 5 draws conclusions.

2. Prior work

We categorize object correspondence approaches into *supervised* and *unsupervised* algorithms. Supervised techniques depends either upon the information contained in training samples or supplied manually by users. Several authors have proposed supervised association approaches such as Kettnaker *et al.* [6], Huang *et al.* [3], Dick *et al.* [1] and Wang *et al.* [9]. Unlike supervised techniques, unsupervised techniques do not require training samples or manual selection of the parameters. Recent unsupervised target as-

[*] This work was supported in part by the EU, under the FP7 project APIDIS (ICT-216023).

978-0-7695-3718-4/09 $25.00 © 2009 IEEE
DOI 10.1109/AVSS.2009.65

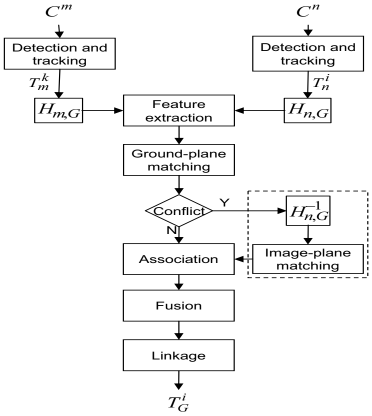

Figure 1. Flow diagram of the proposed approach.

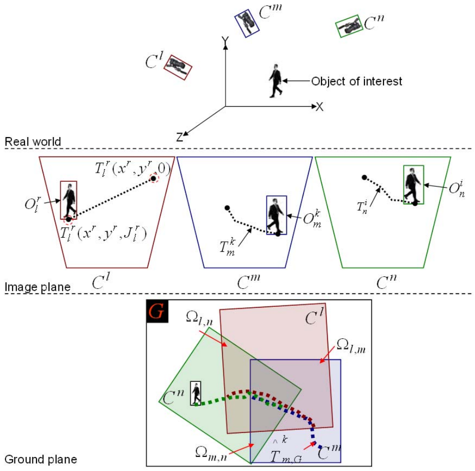

Figure 2. Illustration of notations.

sociation algorithms are presented by Kayumbi *et al.* [4] and Sheikh *et al.* [7]. The rest of this section provide the details of these techniques.

Kettnaker *et al.* [6] presented a Bayesian solution to track people across multiple cameras. The system requires prior information about the environment and the way people move across it. Huang *et al.* [3] presented a probabilistic approach for tracking cars across two cameras on a highway, where transition times were modeled as Gaussian distributions. Like Kettnaker *et al.*, it was assumed that the initial transition probabilities were known. This approach is application-specific, using only two calibrated cameras with vehicles moving in one direction in a single lane. Dick *et al.* [1], use a stochastic transition matrix to describe patterns of motion for both intra- and inter-camera correspondence. The correspondence between cameras has to be supplied as training data. Wang *et al.* [9] connect trajectories observed in multiple cameras based on their temporal information. The trajectories are considered to be corresponding, if they overlap in time for a empirically pre-selected interval.

Kayumbi *et al.* [4] establish correspondence between cameras and virtual ground-plane. Trajectory association is done on the ground-plane using shape and length along with temporal information. The maximum likelihood for association is calculated by cross correlation of spatio-temporal feature vectors. However, this approach cannot differentiate two objects moving with varying speed in the environment. Another approach in this category is presented by Sheikh *et al.* [7], in their approach, airborne cameras are used with the assumption of the simultaneous visibility of at least one object by two cameras. Taking as input time-stamped trajectories from each view, the algorithm estimates the inter-camera transformations. The maximum likelihood is estimated as a function of the reprojection error. A pair of trajectories is considered as generated from the same object if the reprojection error is minimum.

3. Global trajectory generation

3.1. Problem formulation

Let $C = \{C^1, C^2, ..., C^N\}$ be a set of N partially overlapping synchronized cameras (Fig. 2). Let O_n^i represent the i^{th} object observed in C^n. We perform video object extraction (foreground segmentation) using a statistical color change detector and then we associate them across consecutive frames using graph-matching [8]. Let $T_n^i(x^i, y^i, t)$ be the resulting observation (track-point) of O_n^i on location (x^i, y^i) at instant t in camera C^n. The trajectory of O_n^i is a set of all observations i.e., $\mathbf{T}_n^i = \{T_n^i(x^i, y^i, 0), T_n^i(x^i, y^i, 1), ..., T_n^i(x^i, y^i, J_n^i)\}$, where J_n^i represent the length of the trajectory.

T_n is pixel co

We construct a virtual ground-plane (G) from the available information of the datasets and the image-plane to ground-plane projection is estimated by applying the homography matrix $H_{n,G}$ [2] i.e.,

$$\hat{T}_{n,G}^i(\hat{x}^i, \hat{y}^i, t) = H_{n,G} T_n^i(x^i, y^i, t), \tag{1}$$

where, $\hat{T}_{n,G}^i(\hat{x}^i, \hat{y}^i, t)$ is the local ground-plane projection of $T_n^i(x^i, y^i, t)$ and $H_{n,G}$ is the homography matrix. $H_{n,G}$ is constructed by selecting control points to establish the image- and ground-plane correspondence. However, these local projections result in differences in the overlapping region ($\Omega_{m,n}$) on the ground plane. Figure 3(top) shows a network of two partially overlapping cameras and accumulated trajectories in each view (Fig. 3(middle-row)). Figure 3(bottom-left) shows the local projections of the trajectories on a common ground plane, where there are considerable differences of an object's trajectory viewed in two cameras (Fig. 3(bottom-right)). This leads to the requirement of a process which can establish a proximity matrix to associate every p^{th} trajectory to all q^{th} trajectories in $\Omega_{m,n}$. The final goal is to reconstruct a complete global trajectory

my main co

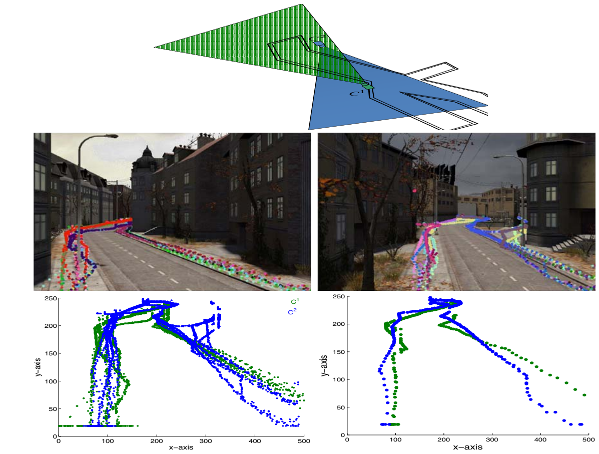

Figure 3. A network of partially overlapping cameras. (Top) configuration of the cameras; (middle-row) accumulated trajectories in each view; (bottom-left) ground-plane projection from the two views and (bottom-right) an example of the differences on the ground-plane.

of an object by fusing the trajectory segments across the entire environment.

3.2. Trajectory association

In the case of partially overlapping cameras, we need to establish the correspondence between transformed trajectory segments ($\hat{\mathbf{T}}^i_{n,G}$) in the overlapping regions on the ground-plane. To find the relative pair-wise similarities for association, we use both spatial and temporal features extracted from the trajectory segments. We assume that a pair of trajectories from different cameras has to be similar both in time and space for the association and fusion. In [4], the shape ($\beta^i_{n,G}$), which is approximated by polynomial coefficients, and length ($\mathbf{d}^i_{n,G}$) are used to find the similarity. However, these features are not generalized enough to handle variety of trajectories. Figure 4(left) shows an example, where two trajectories are considered as similar using these features. In fact, the second object is moving twice the speed of the first one. We expand the feature set by including the average target velocity, $\bar{\mathbf{v}}^i_{n,G}$, which helps in describing the rate of change of the i^{th} object position and is calculated as

$$\bar{\mathbf{v}}^i_{n,G} = \frac{1}{J^i_n} \sum_{j=1}^{J^i_n - 1} \big(\hat{x}^i(j+1) - \hat{x}^i(j), \hat{y}^i(j+1) - \hat{y}^i(j)\big). \tag{2}$$

However, $\bar{\mathbf{v}}^i_{n,G}$ defines the average rate of change of an entire trajectory segment. For localizing (time and position) the abrupt changes in a trajectory, we employ the sharpness of turns ($\hbar^i_{n,G}$), which defines the statistical directional characteristics of a trajectory and is calculated as

$$\hbar^i_{n,G} = H(\theta^i_{n,G}), \tag{3}$$

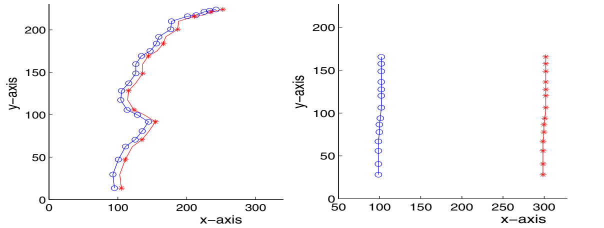

Figure 4. Illustration of feature's limitations. (Left) shape and length features unable to distinguish the trajectories belonging to different objects and (right) shape, length and average velocity features unable to differentiate two trajectories that are spatially far from each other.

where $H(.)$ is a histogram function calculated over the directional angles ($\theta^i_{n,G} = tan^{-1}(\hat{y}^i(j+1) - \hat{y}^i(j)/\hat{x}^i(j+1) - \hat{x}^i(j))$. We take the indices to the top three peaks of $\hbar^i_{n,G}$ as they describe the dominant angles in the trajectory. Furthermore, we consider a situation where two trajectories with similar shape, length and velocity are present in completely different regions of the environment (see Fig. 4(right)). In order to distinguish them, trajectory mean ($\mathbf{m}^i_{n,G}$) is used and is defined as

$$\mathbf{m}^i_{n,G} = \frac{1}{J^i_n} \sum_{j=1}^{J^i_n} \big(\hat{x}^i(j), \hat{y}^i(j)\big). \tag{4}$$

The features discussed so far define the overall pattern of a trajectory. In order to get the variation information at the sample level, we include PCA components analysis ($\mathbf{p}^i_{n,G}$). We apply PCA on sample points of each trajectory by considering the covariance matrix as

$$\Xi^i_{n,G} = \frac{1}{J^i_{n,G}} \widetilde{T}^i_{n,G} \widetilde{T}^i_{n,G}, \tag{5}$$

where $\widetilde{T}^i_{n,G}$ is the mean-shifted version of $T^i_{n,G}$. The eigenvalue decomposition of $\Xi^i_{n,G}$ results in eigenvalues, $\alpha = \{\alpha_j\}_{j=1}^{J^i_n}$, and corresponding eigenvectors, $\varphi = \{\varphi_\mathbf{j}\}_{j=1}^{J^i_n}$. After sorting α in descending order, we consider first two $\varphi_\mathbf{k}, \varphi_\mathbf{l} \in \varphi$, corresponding to the top two eigenvalues, $\alpha_k, \alpha_l \in \alpha$, as most of the variation lies in these two components. The final (normalized) feature vector is:

$$\mathbf{\Theta}^i_{n,G} = (\beta^i_{n,G}, \mathbf{d}^i_{n,G}, \bar{\mathbf{v}}^i_{n,G}, \hbar^i_{n,G}, \mathbf{m}^i_{n,G}, \mathbf{p}^i_{n,G})^T, \tag{6}$$

where T denotes the transpose operator. Because of its robustness to the scale variation, we use cross correlation as *proximity measure*. For $\mathbf{T}'^i_{n,G}$ and $\mathbf{T}'^k_{m,G}$ in $\Omega_{n,m}$ the association matrix is calculated as:

$$A_\Omega(\hat{T}^i_{n,G}, \hat{T}^k_{m,G}) = \varsigma(\mathbf{\Theta}^i_{n,G}, \mathbf{\Theta}^k_{m,G}), \tag{7}$$

where, ς is the correlation function. A trajectory $\hat{\mathbf{T}}^i_{n,G}$ will be associated to any trajectory $\hat{\mathbf{T}}^k_{m,G}$ for which it has max-

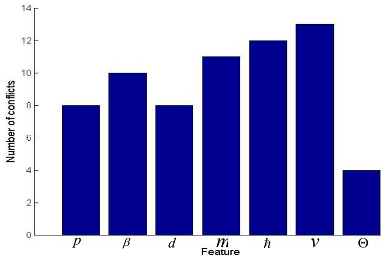

Figure 5. Number of conflicts in individual feature spaces.

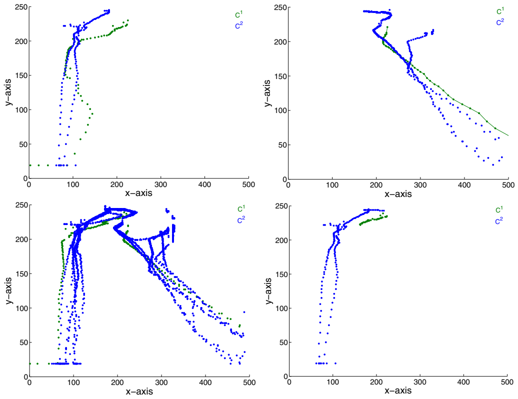

Figure 6. Examples of conflicting situations in (top-left) p, (top-right) m, (bottom-left) combined v and $\hbar$ and (bottom-right) d.

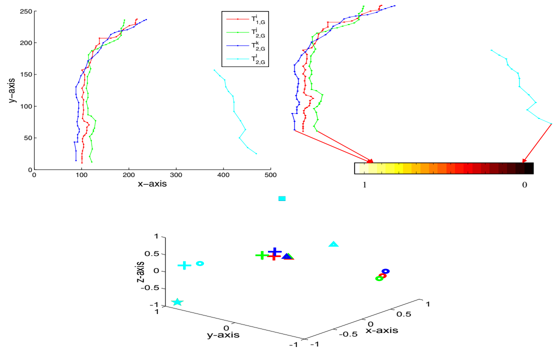

Figure 7. An example of association conflict. (Top-left) four sample trajectories (see Fig. 3); (top-right) association results with two trajectories (blue and green) have equal scores; (bottom) representation of trajectories in each feature space with a marker color maps to trajectory color (key; star:$\bar{\mathbf{v}}$; cross:$\mathbf{m}$; triangle: β; circle: $\mathbf{p}$; square: $\hbar$); particular to $\bar{\mathbf{v}}$ and $\hbar$ all trajectories coincide.

imum correlation i.e.,

$$D_{\Omega} = \arg\max_{r}(A_{\Omega}(\hat{T}^{k}_{l,G}, \hat{T}^{r}_{m,G}))\ \forall\ O^{r}_{m} \in C^{m}. \qquad (8)$$

It is noticed that individual features result in spurious associations as shown in Figure 5. There are number of conflict situations (examples are shown in Fig. 6), however, the use of combined feature set reduces this number considerably. However, still there are cases where multiple trajectories can correspond to a trajectory. Figure 7 demonstrates such an example, where two out of three trajectories match the input trajectory. In order to have single trajectory segment belong to a physical object in the overlapping region, we need to resolve this conflict situation. For this, we perform matching in image-plane by reprojecting the matched trajectories from the ground-plane. Suppose, trajectory segment $\hat{\mathbf{T}}^{i}_{n,G}$ can be associated to $\hat{\mathbf{T}}^{k}_{m,G}$ and $\hat{\mathbf{T}}^{s}_{m,G}$ (or even more), we reproject the trajectories onto the image-plane using $H^{-1}_{n,G}$. Resampling is done in order to have equal length trajectories and then standard Euclidean distance (d) is employed as proximity measure. The trajectory is selected for which the distance is minimum i.e.,

input traj: Ti_1

$$K = \arg\min_{l}(d(\hat{T}^{i}_{n,G}, \hat{T}^{l}_{m,G}))\,\forall\, l = 1, ..., L, \qquad (9)$$

where L is the total number of matched trajectories on the ground-plane.

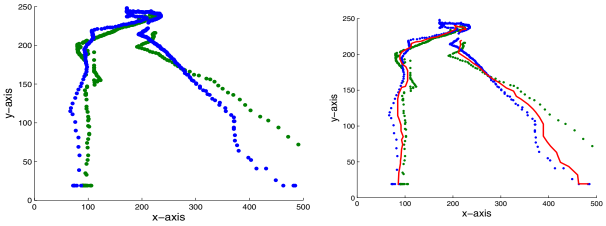

Figure 8. An example of trajectory fusion. (Left) matched trajectories from a pair of overlapping cameras (Fig. 3) and (right) Fusion result.

3.3. Trajectory fusion

Once association is done, the next step is to fuse a pair of corresponding trajectories in overlapping regions. To fuse $\hat{\mathbf{T}}^{i}_{n,G}$ and $\hat{\mathbf{T}}^{k}_{m,G}$, where both trajectories are generated from the same object in real world, we use an adaptive weighting method i.e.,

$$\hat{T}^{i,k}_{n,m,G}(t) = \begin{cases} w_1\hat{T}^{i}_{n,G}(t) + w_2\hat{T}^{k}_{m,G}(t) & in\, R_{n,m} \\ \hat{T}^{i}_{n,G}(t) & in\, R_{n} \\ \hat{T}^{k}_{m,G}(t) & in\, R_{m}, \end{cases} \qquad (10)$$

where $R_{n,m}$ is the region where observations from both $\hat{T}^{i}_{n,G}$ and $\hat{T}^{k}_{m,G}$ are available at t. R_n and R_m are the regions where the observation is available from either $\hat{T}^{i}_{n,G}$ or $\hat{T}^{k}_{m,G}$, respectively. At each time t when the observation from both trajectories are available, the trajectory segment which has more track points is given higher weight than the other; otherwise, we utilize the available observation from one of the trajectories. The weights ($w_i : i = 1, 2$) are calculated as function of number of observations for each trajectory:

$$w_1 = \frac{|\hat{T}^{i}_{n,G}|}{|\hat{T}^{i}_{n,G}| + |\hat{T}^{k}_{m,G}|}, w_2 = \frac{|\hat{T}^{k}_{m,G}|}{|\hat{T}^{i}_{n,G}| + |\hat{T}^{k}_{m,G}|}, \qquad (11)$$

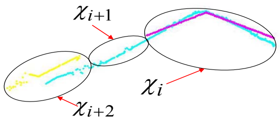

Figure 9. An example of trajectory linkage.

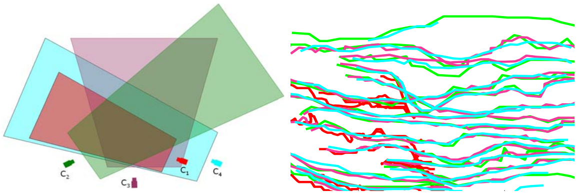

Figure 10. Trajectory segments accumulated over 500 frames from the 4 partially overlapping cameras. (Left) configuration of the cameras; (right) accumulated trajectory segments (segment's color corresponds to the camera's color).

where $|.|$ is the number of observations in a trajectory and w_1+w_2=1. In order to have smoother overall trajectory to avoid small fluctuations due to the observation's gaps, we apply a moving average approach where window size is set to 5 observations.

Finally, to construct a complete trajectory across the entire environment, we connect all segments that belong to same object (see Fig. 9) i.e.,

$$T_G^i = \bigcup_{i=1}^{\kappa} \chi_i, \tag{12}$$

where κ is total number of connected regions. Also, χ_i is the segment observed in overlapping region between C^n and C^m and χ_{i+1} is the segment observed in non-overlapping region (i.e. only in C^m) and χ_{i+2} is the segment observed in overlapping region between C^m and C^l. In this way, a complete trajectory is constructed for cameras C^l, C^m and C^n, whereby C^l and C^n are non-overlapping by configuration.

4. Experimental results

We evaluate the performance of the proposed approach on two real world datasets. The first dataset ($S1$) is an indoor basketball video sequence, which consists of 500 frames (RGB 24 bit images at 25 frames/sec and 1200x1600 pixels), describing a scene simultaneously recorded by 4 cameras located at different viewpoints (see Fig. 10). The second dataset ($S2$) is a more complex soccer match footage, which consists of 3000 frames (RGB 24 bit images at 25 frames/sec and 1920x1080 pixels), describing a scene simultaneously recorded by 6 cameras located at different viewpoints (see Fig. 11). In both datasets, the closeness of players' movement and similarity in team colors make the association task even more challenging. When acquiring these sequences, no constraints were imposed on objects' trajectories. Figure 12 and Fig. 13 show the complete global trajectories of all the objects for both sequences. For both datasets, we used visual data to generate the ground truth for association, we perform objective evaluation of association and fusion results using *Recall* (R) and *Precision* (P). R is the fraction of accurate associations to the true number of associations. P is the fraction of accurate associations to the total number of achieved associations. Let ξ_Ω be the ground truth for pairs of trajectories on the overlapping region Ω and let E_Ω be the estimated results. Then R and P are calculated as:

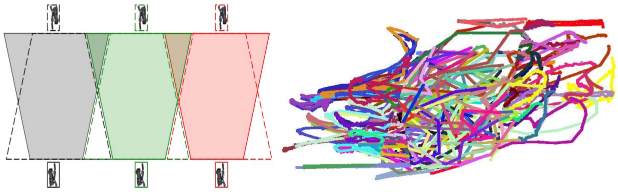

Figure 11. Trajectory segments accumulated over 3000 frames from the 6 partially overlapping cameras of Figure 3. (Left) configuration of the cameras; (right) trajectory segments (color-coded). Note the visibility of the limits of the fields of view of each camera.

$$R = \frac{|\xi_\Omega \cap E_\Omega|}{|\xi_\Omega|}, \tag{13}$$

$$P = \frac{|\xi_\Omega \cap E_\Omega|}{|E_\Omega|}, \tag{14}$$

where $|.|$ is the cardinality of a set.

We compare the performance of the proposed approach with standard Dynamic Time Warpping (DTW) [5] (*M1*) and two state-of-the-art approaches presented in [4] (*M2*) and [7] (*M3*) in terms of P and R for both sequences. The results are compiled in Table 1. The results show that the proposed approach is better by 21% and 18% in R and P. This implies that in complex datasets such as $S1$ and $S2$, where objects are very close in time and space, trajectory statistics help in better association. Furthermore, on average the proposed approach is better by 8% and 6% for R and P respectively, compared to *M2*. Compared to *M3*, the proposed approach outperforms it by 7% and 4% for R and P, respectively. In particular, for dense segments' regions like $\Omega_{1,2}$, $\Omega_{3,4}$ and $\Omega_{3,4}$ in $S2$, it outperformed the other two approach because of the more generic feature-set used built-in verification method. On the other hand, *M2* covers very limited features and lacking a procedure to resolve conflict situations. This results in lower P and R scores. Similarly, if the segments are too close on the image-plane, they cannot be separated using the reprojection error criterion. Therefore, *M3* fails to distinguish the segments that in fact belong to different physical objects exhibiting similar

Table 1. Evaluation and comparison of trajectory association results on $S1$ and $S2$.

	$S1$		$S2$															
	$\Omega_{1,2,3,4}$		$\Omega_{1,2}$		$\Omega_{3,4}$		$\Omega_{5,6}$		$\Omega_{1,3}$		$\Omega_{2,4}$		$\Omega_{3,5}$		$\Omega_{4,6}$		**Average**	
Algorithm	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P
M1	.75	.83	.60	.70	.76	.85	.73	.78	.71	.87	.72	.76	.68	.80	.73	.71	**.71**	**.79**
M2	.95	.93	.80	.90	.88	.90	.81	.98	.90	.90	.82	.96	.82	.90	.78	.81	**.84**	**.91**
M3	1.00	1.00	.76	.81	.81	.92	.78	.99	.89	.92	.84	.97	.87	.96	.82	.85	**.85**	**.93**
Proposed	1.00	1.00	.96	.98	.95	1.00	.96	1.00	.93	.95	.85	.98	.87	.98	.82	.85	**.92**	**.97**

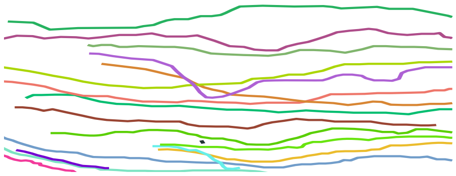

Figure 12. Trajectory association and fusion results across the cameras of $S1$. Each complete trajectory is shown with different color.

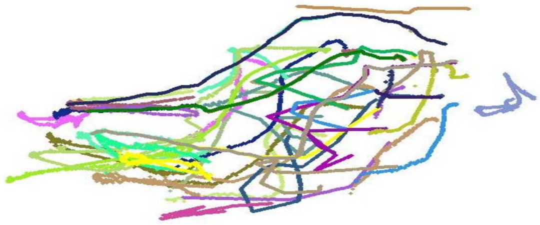

Figure 13. Trajectory association and fusion results across the cameras of $S2$. Each complete trajectory is shown with different color.

motion patterns. However, for lower density regions (see $S1$) both the proposed approach and *M3* produces similar results and outperform *M2*. The results show that on these real world dataset, the proposed approach works accurately for association in both dense and sparse regions.

5. Conclusions

We addressed the problem of trajectory association across partially overlapping cameras in an unsupervised fashion, without imposing constraints on the camera placement. Local trajectory segments from each camera are projected on a common ground-plane. Multiple spatio-temporal features are then analyzed to find the degree of proximity among the trajectories. The matching is verified via ground-plane to image-plane reprojections.

The proposed approach generates a complete trajectory belonging to a physical object in an unsupervised way without requiring learning (a computationally complex process) of motion parameters. We tested the performance of the proposed approach on two real world datasets and found that it outperforms by at least of 4% in precision and 8% in recall state-of-the-art approaches.

Our current work includes employing application-domain information to identify and understand the events of interest from common patterns.

References

[1] A. R. Dick and M. J. Brooks. A stochastic approach to tracking objects across multiple cameras. *Book Series Lecture Notes in Computer Science*, 3339/2005:160–170, Nov. 2004. 1, 2

[2] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second ed. Cambridge University Press (UK), 2004. 2

[3] T. Huang and S. Russell. Object identification in a bayesian context. In *Proc. of International Joint Conference on Artificial Intelligence*, Nagoya (Japan), Aug. 1997. 1, 2

[4] G. Kayumbi, N. Anjum, and A. Cavallaro. Global trajectory reconstruction from distributed visual sensors. In *Proc. of ACM / IEEE Int. Conference on Distributed Smart Cameras*, California (USA), Sep. 2008. 1, 2, 3, 5

[5] E. Keogh. Exact indexing of dynamic time warping. In *Proc. of Intl. Conf. on Very Large Data Bases*, Hong Kong (China), Aug. 2002. 5

[6] V. Kettnaker and R.Zabih. Bayesian multi-camera surveillance. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, Fort Collins, CO (USA), Jun. 1999. 1, 2

[7] Y. Sheikh and M. Shah. Trajectory association across multiple airborne cameras. *Trans. on Pattern Analysis and Machine Intelligence*, 30(2):361–367, Feb. 2008. 1, 2, 5

[8] M. Taj, E. Maggio, and A. Cavallaro. Multi-feature graph-based object tracking. In *CLEAR, Springer LNCS 4122*, pages 190–199, Southampton (UK), Apr. 2006. 2

[9] X. Wang, K. Tieu, and W. Grimson. Correspondence-free multi-camera activity analysis and scene modeling. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Alaska (USA), Jun. 2008. 1, 2