What Epipolar Geometry Can Do for Video-Surveillance

Nicoletta Noceti, Luigi Balduzzi, and Francesca Odone

DIBRIS - Università degli Studi di Genova
via Dodecaneso, 35 - 16146-IT, Genova
{Nicoletta.Noceti,Luigi.Balduzzi,Francesca.Odone}@unige.it

Abstract. In this paper we deal with the problem of matching moving objects between multiple views using geometrical constraints. We consider systems of still, uncalibrated and partially overlapped cameras and design a method able to automatically learn the epipolar geometry of the scene. The matching step is based on a functional that computes the similarity between objects pairs jointly considering different contributions from the geometry. We obtain an efficient method for multi-view matching based on simple geometric tools, requiring a very limited human intervention, and characterized by a low computational load. We will discuss the potential of our approach for video-surveillance applications on real data, showing very good results. Also, we provide an example of application to the consistent labeling problem for multi-camera tracking, and report a comparative analysis with other methods from the state of the art on the PETS 2009 benchmark dataset.

Keywords: Epipolar geometry, multi-view object tracking, video-surveillance.

1 Introduction

State-of-the-art video-surveillance systems available on the market often adopt multiple cameras to be able to monitor large environments and tackle complex situations [1]. Quite surprisingly, the algorithms processing the acquired video streams rarely exploit prior information on the systems geometry. On one hand, in minimal configuration setups cameras have a small or null overlap, and thus system calibration becomes difficult and often not enough reliable. On the other hand, redundant setups are characterized by large field of views overlaps, making the calibration process more reliable but time-consuming. Also, all calibration procedures usually require a high degree of intervention of specialized users and may be not always accepted by surveillance systems installers.

In this work we consider systems of still, partially overlapped and uncalibrated cameras, observing generic environments with a moderate crowding level. Our goal is to build a model of the overall scene dynamics evolving over time. The method we propose is based on a coarse annotation of the scene, that identifies the main walkable components, as ground floor and stairs, that we approximate

A. Petrosino (Ed.): ICIAP 2013, Part I, LNCS 8156, pp. 442–451, 2013.

with planes. The annotation is the only part in the whole pipeline requiring human intervention. Given a pair of overlapped cameras, we relate the scenes at a global level – estimating the fundamental matrix – and at a more local one, building a homography relationship between each pair of homologous regions. Global and local geometrical constraints jointly contribute – possibly with different weights – to the evaluation of the similarity between objects observed in different views. Computing the pairwise similarity between all the objects at a given time t, we populate a matrix from which we deduce matching relationships and missing elements.

Over the last decades several methods adopting geometry within multi-camera systems have been proposed. Among the first attempts, the work in [12] addresses the problem of self-calibration of multiple cameras using feature correspondences to determine the camera geometry. It assumes planarity of the observed scene and sets the basis for working with an overhead view. In [14,17,19] calibration is used to model the 3D relationships between overlapped cameras. However, in most cases full calibration is not available, thus geometry is recovered estimating geometrical transformations between the views from image features. Multi-object matching has been addressed by imposing or learning geometrical constraints on the observed scene [13,2,6,4,16,3,9], often assuming planarity of the ground [16,3,9,13]. Some methods propose to precisely estimate the boundaries of the Field Of View (FOV) to disambiguate among the multiple possible objects associations [3,9], other methods tackle the same problem by combining geometry with appearance models [5,4].

The main contribution of this work is an analysis of what well-accepted geometrical tools can do to improve the reliability of real video-surveillance systems. The result is a method that, given a coarse annotation of the scene geometry, first provides a viable calibration procedure, and second builds a model of the scene dynamics which we use to match objects across the views. This model could be applied to deal with occlusions, tracking noise and consistent labeling. We do not add constraints on the environments and do not need to explicitly determine the common fields of view. Our method requires a very limited human intervention and is computationally very efficient, providing real-time performances.

We experimentally evaluate the multi-camera matching *per se* – addressing the so-called consistent labeling problem [9] – on both annotated data and observations obtained from a tracker to discuss the potential of our solution. Then, we also evaluate the accuracy of our method within multi-camera tracking on the benchmark data of *PETS 2009*, comparing them with the state of the art.

The rest of the paper is organized as follows. In Sec. 2 we discuss the estimation of the geometry, while in Sec. 3 we detail our approach. Sec. 4 and 5 are devoted to the experimental analysis, and, finally, Sec. 6 is left to discussions.

2 Estimation of the Geometry between Two Views

In this section we discuss a simple way to estimate the epipolar geometry between camera pairs, specifically designed to be practicable for video-surveillance systems installations.

The correspondences between the views can be easily established by considering videos with a single person walking, spanning all the walkable floor regions of the scene, similarly to [3]. We employ a motion segmentation algorithm to locate the moving objects and extract two points for each object: the head, or *upper point* (**up**), and the feet, or *lower point* (**lp**). We show in Fig. 1 an example of input for a given camera pair (upper points in green, lower points in blue). We thus finally collect the ordered sets $UP^c = \{\mathbf{up}_k^c\}_{k=1}^K$ and $LP^c = \{\mathbf{lp}_k^c\}_{k=1}^K$, where $c = \{1, 2\}$ from now on refers to the camera index and K is the number of corresponding points. We also call P^c the union of lower and upper points for each camera: $P^1 = UP^1 \cup LP^1$, $P^2 = UP^2 \cup LP^2$.

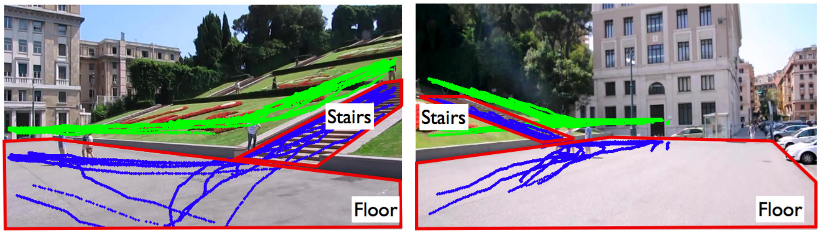


Fig. 1. An example of input corresponding points (upper points in green, lower points in blue). Two main regions are annotated, which correspond to ground floor and stairs.

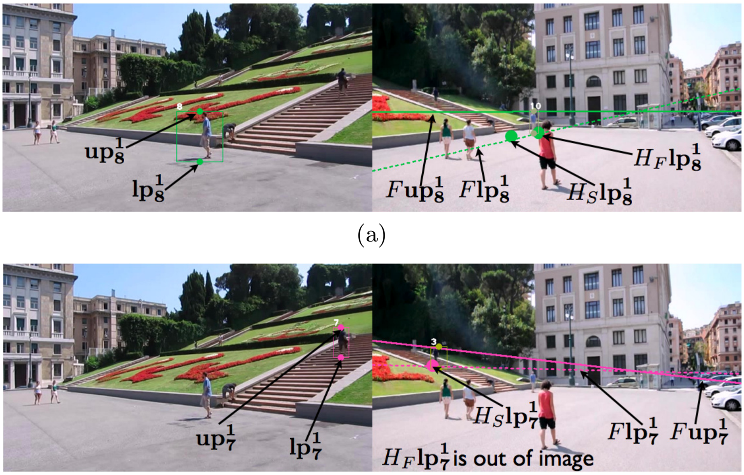
To cope with possible non-planarities of the scene, each image plane is coarsely manually annotated to identify the main structural elements. We only consider walkable, not occluded regions (see Fig. 1) that may be approximated with a plane and characterized by a significant spatial extent.

Let us assume $\mathbf{R} = \{R_n\}_{n=1}^N$ is the set of regions globally annotated in the two cameras. We first consider the transformation between the image planes as a whole, that is the fundamental matrix F: $(\mathbf{p}^2)^T F \mathbf{p}^1 = 0$ with $\mathbf{p}^1 \in P^1$ and $\mathbf{p}^2 \in P^2$ corresponding points. Then we focus on the M *regions observed in both the views*, $M \leq N$. For each one of those, we estimate the homography H_m, such that $\mathbf{p}^2 = H_m \mathbf{p}^1$, where $p^1 \in LP^1$ lies on R_m^1 and $p^2 \in LP^2$ lies on R_m^2.

We solve all the obtained systems using the Direct Linear Transformation (DLT) algorithm with RANSAC [7], that allows us to cope with the presence of outliers (strongly affecting our input data) and avoid unstable solutions. At the end of the calibration procedure, we have obtained the matrices modeling the geometrical transformation from scene 1 to scene 2: F and H_m, $m = 1 \dots M$.

3 Matching across Views

In this section we describe our approach to matching between multiple views. For the sake of clarity, in what follows we consider a single pair of overlapped cameras. In presence of more than two pairwise overlapped cameras, a graph modeling the transitions between cameras can be defined [16,3] to guide the



(a)

(b)

Fig. 2. The different contributions to the similarity measure of a person walking on the ground floor (Fig. 2(a)) and on the stairs (Fig. 2(b)). F is the fundamental matrix, while H_F and H_S refer to the homographies related to ground floor and stairs regions, respectively.

visit of the cameras network. The transitive property of subsequent assignments on different cameras guarantees the global consistency of the labeling.

3.1 Geometry-Based Objects Similarity

Let $O_t^1 = (\mathbf{up}_t^1, \mathbf{lp}_t^1)$ and $O_t^2 = (\mathbf{up}_t^2, \mathbf{lp}_t^2)$ be the descriptors of two objects observed in scene 1 and 2 respectively. We define $d_1 = d(F\mathbf{lp}_t^1, \mathbf{lp}_t^2)$ and $d_2 = d(F\mathbf{up}_t^1, \mathbf{up}_t^2)$, where d denotes the geometric distance between the epipolar lines and a point. Then we introduce the contributions of the regions R_m, $1 \leq m \leq M$ common to the views: $d_m = ||H_m\mathbf{lp}_t^1 - \mathbf{lp}_t^2||_2$.

The similarity between the objects is a combination of all the contributions:

$$S(O_t^1, O_t^2) = w_1 \exp\left(\frac{-d_1^2}{2\sigma^2}\right) + w_2 \exp\left(\frac{-d_2^2}{2\sigma^2}\right) + \sum_{m=1}^{M} w_{3+m-1} \exp\left(\frac{-d_m^2}{2\sigma^2}\right) \quad (1)$$

where σ controls the spatial region in which associations should be considered, while the ws weight the importance of each contribution to the final results. They might be chosen depending on prior information when available, or estimated from the data with an appropriate training procedure.

Notice that our method does not require a precise estimation of the common fields of view. Although we restrict the analysis to common regions only, the areas actually observed by each camera might only partially overlap. Thanks to the use of different geometrical contributions, we are able to automatically cope with points missing in one of the two views. In Fig. 2 we provide two examples of the contributions to the similarity measure.

3.2 Objects Matching

Let us assume to have, at time t, two scenes $S_t^1 = \{O_{i,t}^1\}_{i=1}^{N_1}$ and $S_t^2 = \{O_{j,t}^2\}_{j=1}^{N_2}$. In order to compute the matching between the two scenes, we build a matrix $M \in \mathbb{R}^{N_1 \times N_2}$ where each element is computed as

$$M(i,j) = \frac{S(O_{i,t}^1, O_{j,t}^2) + S(O_{j,t}^2, O_{i,t}^1)}{2}. \qquad (2)$$

This matrix models the dynamics of the scene at time t as observed from the views 1 and 2. Since the number of elements in the scenes can be different, we fix a threshold τ of minimum similarity under which an entry of M is set to zero.

From M we may deduce which are the objects belonging to both the views (matches) and what objects are present in one view only (objects without a match). In the case of noiseless data, we could assume that when an object of scene 1 is viewed also in scene 2 then it will correspond to *exactly* one of its objects. In this case a match could be indicated by an entry in M maximum on its row and column (e.g. through the Hungarian algorithm [10]). Missing elements of scene 1 would be denoted by a null row, missing elements from scene 2 by a null column.

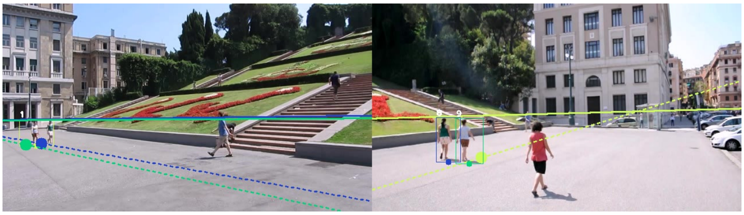


Fig. 3. An example of segmentation error: object 1 of scene 1 corresponds to objects 8 and 9 of scene 2

However in real world applications, the data intrinsically contain noise, due to error in the object segmentation and to objects partially overlapping. Thus, it is very likely to happen that an object of one view actually corresponds to more than one object in the other (see e.g. Fig. 3). We thus modify the matching rule to account for not univocal associations, by imposing that if $M(i,j) > 0$ then it is the only one on its row *or* on its column. If not so, we progressively simplify the matrix by rejecting the lowest values of a subregion of M until the condition holds. The subregion is identified by rows and columns of the elements in the i-row $(M(i,-))$ or in the j-column $(M(-,j))$ greater than zero.

4 Experiments on Multi-camera Matching

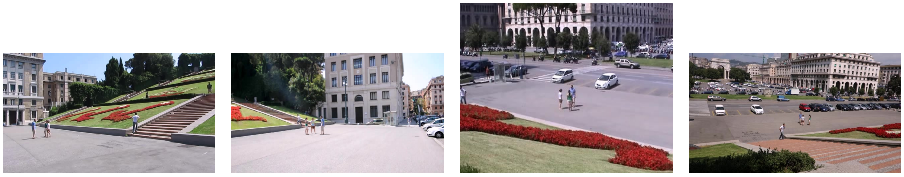
We validated our approach to multi-camera matching on a dataset acquired inhouse (the dataset will be available for download at `http://slipguru.disi.unige.it`). It consists of 4 cameras with partially overlapped views (see Fig.4) that monitor a moderately crowded outdoor environment. All the cameras but *Cam* observe both planar and non-planar areas. A ground truth of the trajectories is available, with a common identifier between all the views for each person.

We annotated ground floor and stairs in each scene and estimated the projection matrices F, H_F and H_S. To evaluate the performance of our approach we interpret the data by considering matched objects as positives and objects without a match as negatives. We take into account both, estimating Positive Predictive Value, True Positive Rate, Negative Predictive Value, True Negative Rate, Accuracy and F-measure. In all the experiments, we fix $\tau = 0.5$ and $\sigma = 15$, while the weights ws are automatically selected on a training set using a grid-search approach (each w is sampled in the range $[0, 1]$ with sampling step 0.05 and such that the weights sum up to 1) and selecting the best performing combination in terms of matching correctness. We use the first minute of each video as training set, the remainder (about $2'$) is adopted as a test set.

4.1 Assessment on Annotated Data

We first assess our approach on the annotated objects. In Tab. 1 are the weights learned automatically from the data, that reflect the peculiarities of the scenes pair. In the case of the pair *Ixus*-*Cam*, e.g., a great importance is done to H_F, since *Cam* only observes the floor. The homography is also rather significant for *Nikon*-*Cam* even if the distance between the common observed floor region and *Nikon* makes the contribution of H_F more noisy and thus the measures obtained with F gain a higher weight. When the cameras observe ground floor and stairs both the homographies are taken into account and enforced with at least one contribution from the epipolar geometry.

Tab. 2 reports the comparison of the methods with weights learning (last row) with a selection of other configurations. The case $H_F\mathbf{lp} + H_S\mathbf{lp}$ and $F\mathbf{lp} + F\mathbf{up}$ are considered as minimal configurations that explicitly account for scene with multiple planar regions. As shown, the weights learning allows us to reach the best results, showing the capability of adapting to different scene peculiarities.



(a) *Ixus* (b) *Nikon* (c) *Cam* (d) *Rx100*

Fig. 4. Example frames of the dataset we adopted in the experimental analysis

Table 1. Weights selected via the training stage

Camera pair	**W**			
	F**lp**	F**up**	H_F**lp**	H_S**lp**
Ixus-Nikon	0	0.6	0.2	0.2
Ixus-Cam	0	0.2	0.8	0
Ixus-Rx100	0	0.4	0.1	0.5
Nikon-Cam	0.2	0.3	0.5	0
Nikon-Rx100	0.3	0	0.4	0.3
Cam-Rx100	0	0.5	0.5	0

Table 2. Average performance of the matching procedure over all cameras pairs on annotated data

Configuration	PPV	TPR	NPV	TNR	ACC	F
H_F**lp**	**0.98**	0.74	0.67	0.99	0.80	0.79
H_S**lp**	0.94	0.56	0.40	**1.00**	0.67	0.65
F**lp**	0.76	0.72	0.68	0.93	0.81	0.78
F**up**	0.62	0.57	0.54	0.86	0.69	0.64
H_F**lp** + H_S**lp**	0.96	0.74	0.54	0.99	0.80	0.79
F**lp** + F**up**	0.79	0.73	0.68	0.92	0.82	0.79
Our approach	**0.98**	**0.96**	**0.91**	0.99	**0.97**	**0.97**

4.2 Evaluations on Measured Data

We now move to the analysis on real data. At each time instant we first apply a motion-based object segmentation and update a tracking [15] on each view independently, and then run the multi-view matching. We consider a match as correct if the two objects involved correspond to the same identity in the ground truth. The correspondence might be partial because of not univocal associations. Similarly, an object of one scene is considered correctly not matched if in the other view it is not observed or a detection is missing for it. The average performances (Tab. 3) show very accurate results although a decrease in the

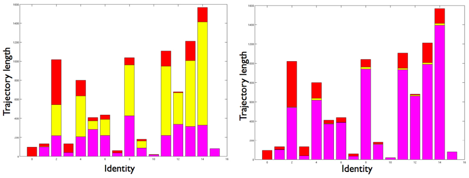
Table 3. Average performance of the matching procedure over all cameras pairs on measured data

Configuration	PPV	TPR	NPV	TNR	ACC	F
H_F**lp**	0.61	0.58	0.76	0.79	0.71	0.71
H_S**lp**	0.36	0.21	0.67	**0.94**	**0.77**	0.73
F**lp**	0.45	0.80	0.88	0.54	0.60	0.64
F**up**	0.36	0.77	0.86	0.40	0.49	0.53
H_F**lp** + H_S**lp**	**0.66**	0.57	0.78	0.82	0.75	0.75
F**lp** + F**up**	0.42	**0.87**	**0.91**	0.43	0.54	0.58
Our approach	0.65	0.84	**0.91**	0.73	0.76	**0.79**

values due to the noise in the data. In this gap we can read an intrinsic limit of this approach: it is expected to increase as the crowd level in the scene grows, influencing the matching accuracy.

5 Experiments on Multi-camera Tracking

To show a possible application for video-surveillance, we apply our matching strategy to assign a common identifier to the same person observed from different views. This problem is commonly referred to as *consistent labeling*. At each time instant, we consider the tracking history from each camera and apply the multi-view matching. If for a given time interval a match has been continuously detected we assign a common identifier to the two objects and label the match as stable. Then, we exploit the association to recover the identities of objects whose trajectory has been cut in subparts, due to tracking failures.



(a) Single-camera (b) Two-cameras

	Ixus	*Nikon*	*Cam*	*Rx100*
Ixus	0.61	0.98	0.97	0.97
Nikon	0.98	0.45	0.96	0.98
Cam	0.99	0.99	0.67	0.98
Rx100	0.95	0.95	0.94	0.48

Fig. 5. Tracking results using 2 cameras observations to compute consistent labeling

We first consider our dataset and show in Fig. 5 the tracking performance for a cameras pair (*Ixus-Nikon*, the one in Fig. 1). Each bar corresponds to an identity in the ground truth, the height reflects the trajectory length. In red we denote the length of the *annotated* trajectory, while in yellow the length of the *measured* one. The latter might be lower due to tracking failures. The magenta bars give a visual impression of the spatio-temporal overlap between trajectories annotated and reconstructed with single-view (Fig. 5(a)) or multi-view (Fig. 5(b)) tracking. The latter clearly allows us to recover from tracking failures.

To show such capability in general, we evaluate the average percentage of spatio-temporal overlap between annotated and reconstructed trajectories. If more than one trajectories correspond to the same annotated object, we only consider the longest. We report the results in table of the right of Fig. 5 (1 means full overlap). The diagonal brings information on single view tracking, while the other values (i,j) tell us how much the i-th camera benefits from the mutual observations with the j-th camera. The table nicely show the gain of multi-view analysis.

We finally evaluate the performances of our approach on the benchmark dataset *PETS 2009*[1]. Due to the limited number of observations, we avoid the

[1] `http://www.cvg.rdg.ac.uk/PETS2009/a.html`

weights learning and instead force the geometrical constraints (F and a single H) to have the same importance. We compare our results on tracking accuracy and missing detections (evaluated following the *CLEAR* metric [8]) with the analysis reported in [11], where the authors propose a method to jointly track multiple objects in multiple views based on formulating the assignment problem as a min-cost problem. Tab. 4 reports the comparison: for camera pairs, our method performs comparably to [11], but differently from [11] we have a gain when increasing the number of cameras to three.

Table 4. Performance evaluation on *PETS 2009* benchmark data. In brackets, the number of cameras adopted for the evaluations.

Method	TA	Miss. Det.
Zhan et al.(1) [18]	**0.66**	0.28
Our approach [15] (1)	0.5	**0.26**
Proposed in [11] (2)	0.76	**0.17**
Our approach (2)	**0.79**	0.19
Proposed in [11] (3)	0.71	**0.13**
Our approach (3)	**0.8**	0.16
Berclaz et al. [1] (5)	0.75	–

6 Discussions

In this paper we showed how very simple geometrical tools can be profitably adopted within multi-camera surveillance setups. We considered systems of still, partially overlapped and uncalibrated cameras and proposed a multi-view matching strategy based on geometrical constraints. Our method estimates the epipolar geometry, and is based on a coarse annotation of the scene. We designed a similarity function making use of different geometry ingredients with variable importances, and showed in the experimental analysis – performed on real data – that learning the weights directly from the data allowed us to automatically adapt to general environment. We reported object matching performances on both annotated and measured data, validating our approach. We finally discussed the potential of our method to address the consistent labeling problem. We compared our method with other state-of-art approaches on the benchmark dataset *PETS 2009*, showing the benefit of increasing the number of cameras.

As a future development, we will integrate the matching module in a real surveillance setting. This will allow us to, on one hand, collect large amount of data, while, on the other, test the robustness of the method with respect to time.

References

1. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. PAMI 33(9), 1806–1819 (2011)
2. Black, J., Ellis, T.: Multi-camera image measurement and correspondence. Measurement 32(1), 61–71 (2002)

3. Calderara, S., Cucchiara, R., Prati, A.: Bayesian-competitive consistent labeling for people surveillance. PAMI 30(2), 354–360 (2008)
4. Chang, T., Gong, S., Ong, E.: Tracking multiple people under occlusion using multiple cameras. In: BMVC, pp. 566–576 (2000)
5. Chang, T.H., Gong, S.: Tracking multiple people with a multi-camera system. In: Work. on Multi-Object Tracking (2001)
6. Dockstader, S., Tekalp, A.: Multiple camera tracking of interacting and occluded human motion. Proc. of the IEEE 89(10), 1441–1455 (2001)
7. Fishler, M., Bolles, R.: Random sample consensus: a paradigm for model fitting with applications to image analtsis and automated cartography. In: Comm. ACM, vol. 24, pp. 381–395 (1981)
8. Kasturi, R., Goldgof, D., Soundararajan, P., Manohar, V., Garofolo, J., Bowers, R., Boonstra, M., Korzhova, V., Zhang, J.: Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. PAMI 31(2), 319–336 (2009)
9. Khan, S., Shah, M.: Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. PAMI 25(10), 1355–1360 (2003)
10. Khun, H.: The hungarian method for the assignment problem. Naval Research Logistic Quarterly 2, 83–97 (1955)
11. Leal-Taixe, L., Pons-Moll, G., Rosenhahn, B.: Branch-and-price global optimization for multi-view multi-target tracking. In: CVPR, pp. 1987–1994 (2012)
12. Lee, L., Romano, R., Stein, G.: Monitoring activities from multiple video streams: Establishing a common coordinate frame. PAMI 22(8), 758–767 (2000)
13. Mittal, A., Davis, L.: Unified multi-camera detection and tracking using region-matching. In: Work. on Multi-Object Tracking, pp. 3–10 (2001)
14. Mittal, A., Davis, L.S.: M2Tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 18–33. Springer, Heidelberg (2002)
15. Noceti, N., Destrero, A., Lovato, A., Odone, F.: Combined motion and appearance models for robust object tracking in real-time. In: AVSS (2009)
16. Stauffer, C., Tieu, K.: Automated multi-camera planar tracking correspondence modeling. In: CVPR, vol. 1, pp. I–259 (2003)
17. Yue, Z., Zhou, S.K.: Robust two-camera tracking using homography. In: Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 1–4 (2004)
18. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: CVPR, pp. 1–8 (2008)
19. Zhou, Q., Aggarwal, J.K.: Object tracking in an outdoor environment using fusion of features and cameras. Image Vision Comput. 24(11), 1244–1255 (2006)