Int J Comput Vis
DOI 10.1007/s11263-013-0624-1

Multiframe Many–Many Point Correspondence for Vehicle Tracking in High Density Wide Area Aerial Videos

Imran Saleemi · Mubarak Shah

Received: 21 December 2011 / Accepted: 2 April 2013

Abstract This paper presents a novel framework for tracking thousands of vehicles in high resolution, low frame rate, multiple camera aerial videos. The proposed algorithm avoids the pitfalls of global minimization of data association costs and instead maintains *multiple object-centric* associations for each track. Representation of object state in terms of *many to many* data associations *per track* is proposed and multiple novel constraints are introduced to make the association problem tractable while allowing sharing of detections among tracks. *Weighted* hypothetical measurements are introduced to better handle occlusions, mis-detections and split or merged detections. A two-frame differencing method is presented which performs simultaneous moving object detection in both. Two novel contextual constraints of vehicle following model, and discouragement of track intersection and merging are also proposed. Extensive experiments on challenging, ground truthed data sets are performed to show the feasibility and superiority of the proposed approach. Results of quantitative comparison with existing approaches are presented, and the efficacy of newly introduced constraints is experimentally established. The proposed algorithm performs better and faster than global, 1–1 data association methods.

Keywords Wide area aerial surveillance · Multi target tracking · Multiframe data association · Multiple hypothesis · Multiple candidate tracking · CLIF dataset

1 Introduction

The goal of the work presented in this paper is the detection and tracking of a large number of moving objects observed from a high altitude aerial platform. Detection and tracking of interesting objects has traditionally been a very important area of research in classical computer vision (Yilmaz et al. 2006), but there are several important challenges related to tracking in high resolution multiple camera aerial videos (e.g., Fig. 1) which allow, not tens or hundreds, but thousands of vehicles to be visible in a single mosaic frame. Firstly, high platform altitude results in extremely small object sizes. Lack of color, coupled with small and variable object sizes (∼50 pixels) imply that object appearance and size are not too discriminative for establishing correspondences, thus making the use of intensity histogram intersection, and template correlation based matching difficult. The second major limitation arises due to the fact that background modeling (Stauffer and Grimson 2000) and frame difference based detection methods assume consistent global illumination and high quality of image registration (global platform motion compensation). Changes in illumination and camera gain are common in aerial videos especially in multi-camera sensors, which require explicit brightness and gain equalization. For image registration, direct estimation methods (Mann and Picard 1997) are computationally prohibitive, while feature based registration results in significant residual errors. Small errors in alignment, coupled with artifacts introduced due to interpolation during warping, also result in intensity variations in even the background regions, which makes automatic

Electronic supplementary material The online version of this article (doi:10.1007/s11263-013-0624-1) contains supplementary material, which is available to authorized users.

I. Saleemi (✉) · M. Shah
Department of Electrical Engineering & Computer Science, University of Central Florida, 4000 Central Florida Blvd., Orlando, FL 32816, USA
e-mail: imran@eecs.ucf.edu

M. Shah
e-mail: shah@eecs.ucf.edu

Published online: 24 April 2013

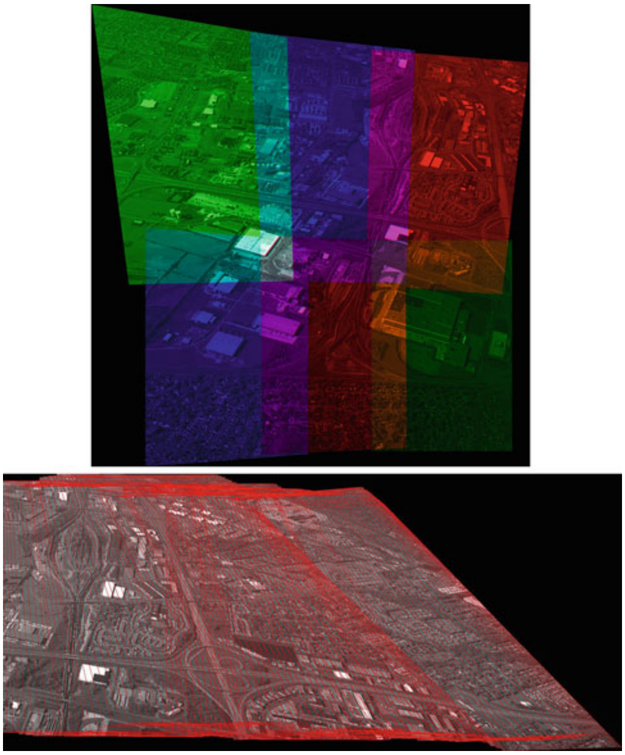


Fig. 1 CLIF dataset (USAF 2006): *Top* mosaic of a six camera array aboard an aerial platform. Mosaic resolution: ∼73 million pixels; each camera's FOV in a different *color* channel. *Bottom single* camera mosaic generated from alignment of 80 frames; boundary of each frame's FOV in *red*. Image resolution: ∼43.7 million pixels (Color figure online)

thresholding of background difference challenging. Moreover, background modeling and accumulative frame difference require significant overlap between frames within temporal windows, a constraint difficult to satisfy in aerial videos due to high platform speeds.

Low frame rate is also a significant source of errors, because object speed *per frame* becomes at least a few times larger than object sizes, so that even with constant velocity dynamics, there are multiple equally likely hypotheses in small temporal windows, while data association criteria like minimization of distance travelled become infeasible. Moreover, since platform motion is much faster than object motion, objects are typically visible only for a few frames, and therefore track initialization (and first correspondence) becomes very important, which, in absence of GIS information, e.g., road orientation, is non-trivial. Finally, heuristics like assumption of constant velocity dynamics for locally proximal objects, which is useful for track initialization, are not valid in general, e.g., for proximal vehicles moving in opposite lanes.

The inspiration for our proposed tracking method is drawn from the simplicity of instantaneous nearest neighbor association methods (NN for single target tracking), and the intuitive benefits of multiple hypothesis tracking (MHT) attributed to Reid (1979). The proposed method however, is significantly different from both, and the similarities and differences are highlighted in Sect. 2. Tracking is a widely studied area of research but given the problems posed by wide area aerial imagery, we propose a new tracking technique whose novel contributions include,

- A probabilistic multiple target tracking framework which unlike conventional methods does not require 1–1 measurement association across frames,
- Maintenance of multiple possible candidate tracks per *object*, rather than multiple *sets* of global 1–1 association configurations,
- Introduction of a new weighted hypothetical measurement derived from observed measurement distribution,
- Application of a vehicle following model for uncongested traffic as additional association constraint, and
- Weighted penalization of track intersection by fast computation of all possible intersections between potential associations.

2 Related Work

The commonly encountered task of multi-target tracking (MTT) can be decomposed into the two coupled problems of state estimation and data association. Numerous motion models and state estimation methods have been proposed in the past for a variety of application scenarios. Examples include constant velocity motion models, and state estimation methods like Kalman filter, and Particle filter (Porikli and Pan 2009; Vezzani et al. 2009; Bazzani et al. 2010). Some proposed methods along with explicit handling of problems like occlusions (Ablavsky et al. 2008), split–merge and entry–exit events (Perera et al. 2006), can perform well for simpler MTT applications. A detailed survey of traditional as well as recent object detection and tracking algorithms has been provided by Yilmaz et al. (2006).

In terms of data association, much of the existing literature on MTT can be categorized into three main classes. These include global nearest neighbor (GNN), joint probabilistic data association filters, and MHT. GNN based methods are a poor choice in the case of low frame rate, high altitude, wide area sequences because most tracks have multiple equally likely hypotheses, and most observations are equally likely to belong to multiple tracks. The reasons for this include very high object density, and the large distances covered by objects between successive observations. One would hope that bipartite graph matching (a type of GNN association) based correspondences (Shafique and Shah 2005) would in general be optimal, and while this is true in a small temporal window, over time, a few wrong correspondences will propagate the error to neighboring object trajectories resulting in

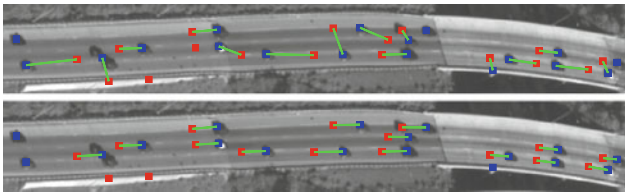


Fig. 2 Examples of *initial* data association for a few objects. Measurements in previous and current frames are shown as *red square*, and *blue square* resp. *Green lines* represent associations. *Top* greedy nearest neighbor initialized with zero velocity, solved globally using Munkres (1957). *Bottom* result using proposed approach (best candidate decided in subsequent frames). Global cost of assignment on *top* is *less* than that on the *bottom*, but incorrect (Color figure online)

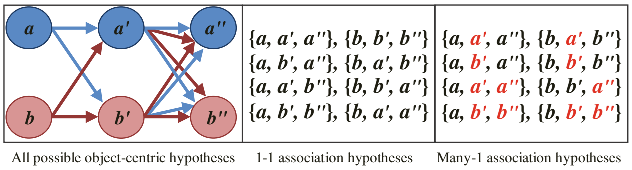

a
a'
a''
b
b'
b''
{a, a', a''}, {b, b', b''}
{a, b', a''}, {b, a', b''}
{a, a', b''}, {b, b', a''}
{a, b', b''}, {b, a', a''}
{a, a', a''}, {b, a', b''}
{a, b', a''}, {b, b', b''}
{a, a', a''}, {b, b', a''}
{a, b', b''}, {b, b', b''}
All possible object-centric hypotheses
1-1 association hypotheses
Many-1 association hypotheses

Fig. 3 Illustration of possible global hypotheses scenarios for a toy example containing two objects in three consecutive frames. *Middle* the four possible *global*, 1–1 associations. Hungarian search (Kuhn 1955) can choose only one of these four as the solution, while MHT (Cox and Hingorani 1996) can propagate all four independently. *Right* many–1 associations by measurement sharing allowed by proposed approach (shared detections shown in *red*) (Color figure online)

wrong matches and corrupting most state estimates, which in general cannot be corrected later (see Fig. 2). Furthermore, bipartite graph matching where each independent set of the bigraph may contain ~1,000 nodes will be computationally expensive.

Joint probabilistic data association filter (JPDAF) is an attractive choice for tracking large number of targets (Shalom and Fortmann 1988), which is an all neighbors association approach, and updates a target using *all* the measurements available near its predicted location, weighted by the motion model based assignment likelihood. This technique essentially is a special, simpler case of MHT, and considers many possible data association hypotheses, but *combines* them after each frame employing a weighted mean of assignment probabilities, instead of propagating individual viable hypotheses. JPDAF is often used in conjunction with Kalman (Kang et al. 2003) and Particle filters (Schulz et al. 2001).

Given the scenario under consideration, it is obvious that detection will be far from perfect, and problems like missed, merged and split detections will be frequently encountered, making instantaneous 1–1 correspondence difficult. Also, the lack of initial velocity estimate requires association deferment for at least a few frames. The only techniques capable of performing association for a large number of objects in a general framework are MHT based methods, which maintain many possible data association hypotheses and propagate the corresponding target state estimates for each hypothesis, essentially deferring correspondence decisions in anticipation that over time with the availability of subsequent data measurements, the joint likelihoods of propagated hypotheses will exhibit increased disparity, thus resolving any ambiguities. Since each hypothesis spawns a new set of child hypotheses at each frame, the MHT approach may result in a combinatorial explosion of hypotheses. However, efficient implementations put forth in Cox and Hingorani (1996), Cox et al. (1997), and Danchick and Newnam (2006), originally due to the result by Murty (1968), allow generation of a best subset of all possible hypotheses, thus making the problem tractable for tracking of a few objects, by bounding both computation and memory.

In order to appreciate the differences between the proposed approach and conventional MHT approaches, it is important to notice that the latter impose a 1–1 correspondence constraint on measurement association, i.e., a single ‘hypothesis’ is a *set* of associations which does not allow 1–many or many–1 correspondence (Fig. 3). Although this constraint is reasonable in many applications (e.g., radar tracking of airborne targets), it can be a severe drawback in the case of high density traffic with extremely noisy measurements. Even relaxing this constraint requires maintenance of a large number of concurrent hypotheses, a fraction of which satisfy viable hypotheses for a *single* object. In other words, enumeration of top 10 candidate tracks for a *single* object may require computation of a much larger set of global hypotheses. Conversely, the top 10 *global* hypotheses may not even have any of the top 10 candidates for a particular object. Furthermore, MHT has much higher computation and memory requirements than even bipartite graph matching, which itself is $O(N^3)$, and given $N = 1{,}000$ is prohibitive.

Another way of analyzing the existing literature in tracking is to consider the types of video considered in them, as well as object density, and resolution, etc. Some existing algorithms have performed well in planar scenes where adequate motion based foreground-background segmentations are achievable (Yin and Collins 2006). Most of the existing methods however, have concentrated on medium and low altitude aerial sequences (Xiao et al. 2008). Although such sequences suffer from stronger parallax induced by out of plane structures, like trees, towers, they offer the important benefit of more pixels per target. In general, many algorithms addressing the problem of MTT have attempted tracking in scenarios involving a maximum of a few tens of objects, e.g., CAVIAR (Yang et al. 2009; Song et al. 2010) (~10 objects per frame, 235 in all), ETHMS (Xing et al. 2009) (maximum of 10 objects per frame), VIVID (Grabner et al. 2008) (less than 5 objects per frame), ETH Central, and soccer and hockey games data sets (Breitenstein et al. 2009) (less than 20 objects per frame). However, with the advent of superior unmanned aerial platforms and cheaper cameras, high resolution, multi-camera, wide area, persistent surveillance

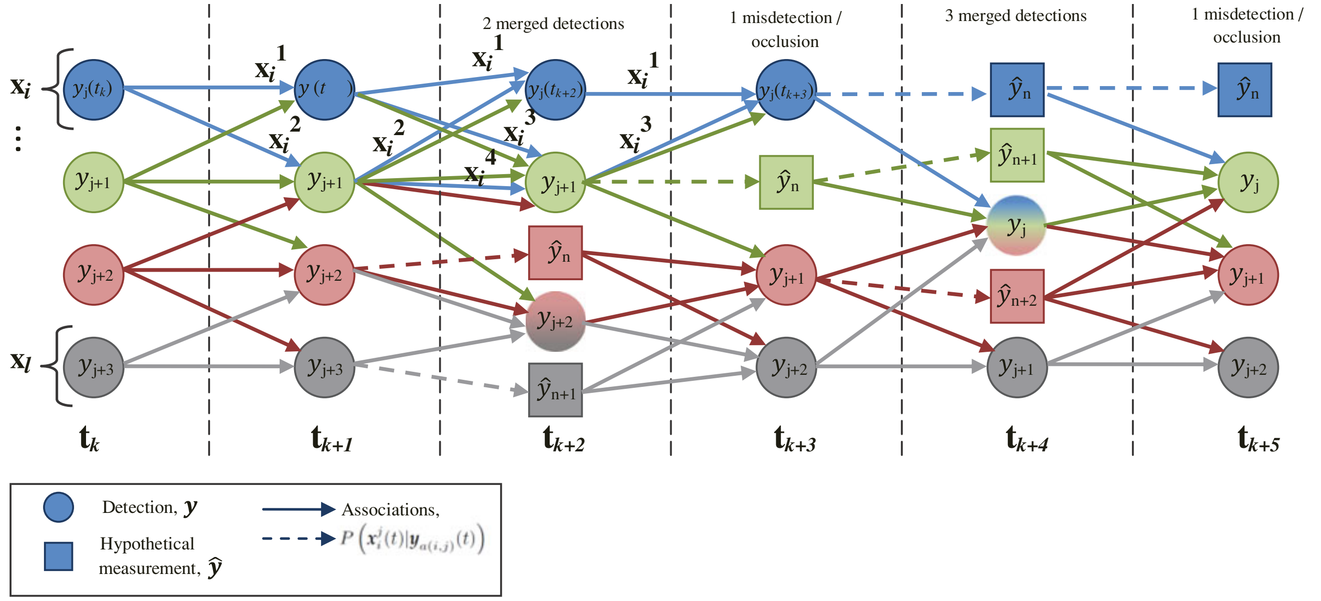


Fig. 4 Overview of the proposed approach. Detections are represented as *circles* or *squares*, separated into columns w.r.t. frames. Each *color* shows detections and associations for different candidates of a single track, while *multicolored circles* depict merged detections. *Distinct colored arrows* between the same pair of detections imply sharing of the detections among tracks. Hypothetical nodes are not shared as indicated by a *single dotted* incoming edge. MTT can be formulated as the problem of finding optimal, disjoint paths in a multi-partite graph. Our proposed solution is to allow paths corresponding to distinct objects to have overlap, by sharing nodes (Color figure online)

is now possible (USAF 2006), where *thousands* of objects may be present in the sensor's FOV. Such sequences have opened doors to new areas of research within the field of object detection, tracking and surveillance.

Recently, some research has been conducted in the area of wide area aerial surveillance (Xiao et al. 2010; Reilly et al. 2010). Both of these methods however perform object association using bipartite graph matching, also known as the Hungarian (Kuhn 1955) or Munkres algorithm (Munkres 1957). The high computation cost of this algorithm, especially for the scenario under consideration, has previously been discussed. Since bipartite graph matching imposes a 1–1 correspondence constraint between objects in consecutive frames, frequently encountered problems of occlusion, mis-detection, detection merging and splitting need to be handled explicitly, which complicate the graph structure in addition to affecting the cost of solution (Shafique and Shah 2005). More importantly, since these methods make correspondence decisions at every frame (instantaneous instead of deferred), additional constraints incorporating the spatial layout of neighboring objects need to be considered, thus causing additional overhead. Given a wrong association in the current frame, the tracker cannot recover, and the mistake is likely to propagate in a cascading fashion. The problem of simultaneous optimization of correspondences across multiple frames has also been attempted using k-shortest path algorithm in Berclaz et al. (2011) and Pirsiavash et al. (2011). While these techniques have achieved impressive results for pedestrian tracking, the k-shortest path algorithm explicitly assumes that tracks (paths in a k-partite graph) are disjoint, thus imposing a 1–1 correspondence constraint on detections in successive frames.

Given this discussion, we now describe the various steps of our approach.

3 Tracking Framework

Our proposed algorithm for association of a large number of measurements across frames deviates significantly from traditional MTT methods (Munkres 1957; Shalom and Fortmann 1988; Cox and Hingorani 1996) by relaxing the 1–1 correspondence constraint as shown in Fig. 4. We observe that in real world data, correct association is often a function of individual object-centric data likelihood and *local* neighborhood context, as opposed to the result of a global cost minimization or likelihood maximization. For example, the knowledge of motion of far away objects in a scene does not affect the cost of associating a particular object. On the other hand, in these methods, existing tracks compete for available measurements, resulting in a *single* winner per measurement, which is an impractically strict constraint on the measurement quality. We present an algorithm for evaluation of a 'many-to-many data association likelihood', i.e., a track can be associated with multiple detections in the next frame (1–many) by retaining multiple candidates per track, and multiple tracks can be associated with a single detection in the next frame (many–1) by measurement sharing (Fig. 4). The proposed method not only maintains multiple possible tracks per object, which we call 'candidates', but also allows candidates from distinct objects to share measurements. We now introduce some notation before formally describing the algorithm:

- **p**: a pixel location in a 2d common image reference frame, i.e., (x, y).
- I_t: image frame observed at time t, aligned to a common reference, $1 \leq t \leq Z$.
- ΔI_t^{t-1}: adaptive consecutive frame difference between images I_{t-1} and I_t.
- $\boldsymbol{x}_i^j(t)$: the *state*, $[\mathbf{p}, \mathbf{v}, \alpha, s]$, of the jth candidate track of the ith object at frame t, where **p** is the current location of the object, **v** and α are the mean velocity and acceleration vectors respectively, while s is the size of the object. The variance σ_s of the object size is also maintained.
- $\Sigma_{i,j}(t)$: a 2×2 covariance matrix of accelerations of the jth candidate of ith object at time t.
- $\mathcal{X}(t)$: set of all existing tracks at frame t, $\{\boldsymbol{x}_i^j | 1 \leq i \leq T_t, 1 \leq j \leq N_i(t)\}$, where T_t is the number of objects observed so far and $T_t \geq T_{t-1}$, and $N_i(t)$ is the number of 'candidate' tracks for the ith object.
- $\mathcal{X}_i'(t)$: set of all existing tracks except $\boldsymbol{x}_i(t)$, i.e., $\mathcal{X}(t) - \{\boldsymbol{x}_i^j(t) | 1 \leq j \leq N_i(t)\}$.
- $\boldsymbol{y}_k(t)$: a measurement vector, $\{(\mathbf{p}_k, \mathrm{s}_k, \mathrm{g}_k)\}$, representing properties of a set of pixels that belong to a moving object k, observed in frame t, where **p** is the object centroid, s is size in pixels, and g is the mean frame difference of the pixels on object.
- $\mathcal{Y}_t$: set of measurements at frame t, $\{\boldsymbol{y}_k(t) | 1 \leq k \leq Q_t\}$.

The proposed framework concentrates on the posterior conditional probabilities of *single* objects, and assumes them to be independent of all but a few other objects. The goal then is to maximize the probability of *each candidate track* as opposed to the joint probability of all tracks (as in Hungarian or MHT). Writing formally, given predictions of existing tracks $\mathcal{X}(t)$ and measurements $\mathcal{Y}_t$ at time t, the goal of our framework is to maintain $N_i(t)$ candidate tracks for object i at time t, where a candidate has the posterior probability, $P\left(\boldsymbol{x}_i^j(t) | \mathcal{X}_i'(t), \mathcal{Y}_{1:t}\right)$. At any given time, the candidate with the highest probability can be picked as the current state of an object as described later. Additionally, a fixed number of top candidates are retained and propagated for subsequent frames. We assume that all candidate tracks of a particular object are mutually independent, while $\boldsymbol{x}_i^j(t)$ and $\mathcal{X}_i'(t)$ are not (since they may share measurements). On the other hand, each of them is dependent on the set of measurements, $\mathcal{Y}_t$. The full posterior probability of a particular candidate therefore involves a recursive relationship with other objects and measurements which results in an intractable computation. We therefore decompose our desired posterior probability for a candidate track into independent and dependent components with respect to the rest of the tracks. The goal is to propagate candidates with the highest values of the following probabilities: P_i, which is the so called object-centric or target-oriented probability, and is independent of not only the rest of the tracks, but also of the rest of the candidates for the object i, and is the subject of Sect. 4; P_f which indicates the probability of association for $\boldsymbol{x}_i^j(t)$ conditioned on all the candidate tracks of all the objects $\mathcal{X}_i'(t)$, that it may be *following* on a road; and P_m which depends on the potential intersection or merging of tracks within short temporal windows, and is high for the candidate $\boldsymbol{x}_i^j(t)$ if it does not intersect with any other candidate of other objects. Only the candidates which intersect with, or are in the spatial vicinity of track $\boldsymbol{x}_i^j(t)$ will affect the terms P_f and P_m. These conditional probabilities will be explored in Sect. 5.

3.1 Logarithmic Opinion Pooling

Given the probabilities from context based cues, as well as the object-centric motion and observation models, the aggregate probability of a candidate is computed by combining them using Logarithmic opinion pooling (Durrant-Whyte 1988), which has been shown to be less scattered than linear pooling (weighted mean). It has also been shown that for the geometric mean of a set of probability distributions, the KL divergence from the true distribution, is smaller than the average of the KL divergences of the individual distributions (Hinton 1999). We assume that the independent cue, P_i, and two novel context aware (dependent) cues, P_f, and P_m are *expert opinions* about the same conditional probability, $P\left(\boldsymbol{x}_i^j(t) | \mathcal{X}_i'(t), \mathcal{Y}_{1:t}\right)$, and are conditionally independent, given $\mathcal{X}_i'(t)$ and $\mathcal{Y}_{1:t}$. We therefore combine these opinions in a weighted fashion as follows:

$$P\left(\boldsymbol{x}_i^j(t) | \mathcal{X}_i'(t), \mathcal{Y}_{1:t}\right) = P_i\left(\boldsymbol{x}_i^j(t) | \mathcal{X}_i'(t), \mathcal{Y}_{1:t}\right)^{\kappa} \cdot P_f\left(\boldsymbol{x}_i^j(t) | \mathcal{X}_i'(t), \mathcal{Y}_t\right)^{\omega} \cdot P_m\left(\boldsymbol{x}_i^j(t) | \mathcal{X}_i'(t), \mathcal{Y}_t\right)^{1-\kappa-\omega}, \tag{1}$$

where $0 < \kappa + \omega < 1$. Each of the terms in the above equation is explained in detail in subsequent sections. Notice that only the object-centric distribution P_i depends on previous measurements ($\mathcal{Y}_{1:t-1}$). The contextual constraints are computed independently for each frame, and only depend on measurements in that frame (in addition to other objects $\mathcal{X}_i'(t)$). This aggregate probability represents the quality of a candidate track, given all the measurements, some of which are shared between this candidate and other tracks, as well as the motion and measurement based likelihoods of the rest of the tracks. We can then choose the best candidate track $\boldsymbol{x}_i^{j^*}(t)$ of an object i, at any given time t, by simply finding the one with maximum aposteriori likelihood, that is,

$$j^* = \underset{j}{\text{argmax}} \quad P\left(\boldsymbol{x}_i^j(t) \middle| \mathcal{X}_i'(t), \mathcal{Y}_{1:t}\right), \tag{2}$$

thereby avoiding the need for any optimization for data-association (bipartite graph matching, MLE, etc.). Examples of the aggregate posterior likelihood of candidates for various tracks are shown in Sect. 7 (Fig. 16).

4 Object-Centric Association

The desired object-centric probability is computed as if the goal were to track only a single object. In other words, the object-centric probability of a candidate track $\boldsymbol{x}_i^j(t)$ is not affected by either other existing object tracks or by measurements other than $\boldsymbol{y}_{a(i,j)}(t)$, where $a(i,j) \in [1, Q_t]$ is the index of the detection that is being associated with $\boldsymbol{x}_i^j(t)$. This probability therefore, depends solely on the motion (process) and detection (observation) models, and is independent of other candidates and tracks. It can be expanded as,

$$\begin{aligned} &P_i\left(\boldsymbol{x}_i^j(t)|\mathcal{X}_i'(t), \mathcal{Y}_{1:t}\right) \\ &\quad = P\left(\boldsymbol{x}_i^j(t)|\mathcal{Y}_{1:t}\right) \\ &\quad = P\left(\boldsymbol{y}_{a(i,j)}(t)|\boldsymbol{x}_i^j(t)\right) P\left(\boldsymbol{x}_i^j(t)|\mathcal{Y}_{1:t-1}\right), \end{aligned} \tag{3}$$

where the second step results from Bayes' rule and the normalizing constant is omitted. Assuming that the distribution $P(\boldsymbol{x}_i^j(t-1)|\mathcal{Y}_{1:t-1})$ is known from the previous frame, the prediction step can be obtained using the Chapman–Kolmogorov equation, and using the fact that $P(\boldsymbol{x}_i^j(t)|\boldsymbol{x}_i^j(t-1), \mathcal{Y}_{1:t-1}) = P(\boldsymbol{x}_i^j(t)|\boldsymbol{x}_i^j(t-1))$ (because first order Markov process):

$$\begin{aligned} P\left(\boldsymbol{x}_i^j(t)|\mathcal{Y}_{1:t-1}\right) = P\left(\boldsymbol{x}_i^j(t)|\boldsymbol{x}_i^j(t-1)\right) \\ P\left(\boldsymbol{x}_i^j(t-1)|\mathcal{Y}_{1:t-1}\right) \end{aligned} \tag{4}$$

The first term in the above equation is based on the motion model, which in our framework is a constant acceleration model, such that at frame t, an existing candidate track $\boldsymbol{x}_i^j(t-1)$ is propagated by predicting it to attain the new state $\hat{\boldsymbol{x}}_i^j(t)$, assuming that the acceleration $\alpha_i^j(t-1)$ is maintained, i.e.,

$$\hat{\mathbf{p}}_i^j(t) = \mathbf{p}_i^j(t-1) + \mathbf{v}_i^j(t-1) + \alpha_i^j(t-1), \tag{5}$$

$$\hat{\mathbf{v}}_i^j(t) = \mathbf{v}_i^j(t-1) + \alpha_i^j(t-1), \tag{6}$$

$$\hat{\alpha}_i^j(t) = \alpha_i^j(t-1). \tag{7}$$

The observation noise function is assumed to be the product of two independent Gaussian distributions and the confidence of the detector in observing a detection, and is written as:

$$\begin{aligned} P\left(\boldsymbol{y}_{a(i,j)}(t)|\boldsymbol{x}_i^j(t)\right) = \mathcal{N}\left(\boldsymbol{y}_{a(i,j)}(t)|\hat{\boldsymbol{x}}_i^j(t), \Sigma_{i,j}(t-1)\right) \cdot \\ \mathcal{N}\left(\boldsymbol{y}_{a(i,j)}(t)|s_i^j(t-1), \sigma_{s,i}^j(t-1)\right) \cdot \mathrm{g}_{a(i,j)}(t), \end{aligned} \tag{8}$$

since an object's motion (acceleration) and size can be assumed to be independent. The detector confidence g is elaborated on in Sect. 4.2.

Using constant acceleration instead of constant velocity motion model has important ramifications in our approach. Although a constant velocity model can handle reasonable variance in velocity, traffic on roads often involves sharp turns like on ramps and intersections, and the wave nature of traffic on congested highways forces abrupt accelerations and decelerations of vehicles, which significantly increases constant velocity based association ambiguity. The benefits of constant angular (yaw) rate and acceleration based motion models for vehicular traffic has been experimentally established in (Schubert et al. 2008).

4.1 Object Detection by Two Frame Difference

Image alignment: Since the aerial platform is in motion, the first step before motion based object detection is the compensation of platform motion, which is performed by detecting Harris corners in consecutive frames, followed by SIFT descriptor computation at each corner, and RANSAC based robust least squares fitting of a Homography transformation (Fig. 1, bottom). While a Homography is computed for all frames w.r.t a reference, only half the frames are actually warped (e.g., even frames warped to preceding odd frames), because the proposed detection method simultaneously performs detection in two consecutive frames. Given overlapping FOVs, inter-camera transformations are similarly estimated, once per sequence (Fig. 1, top).

Why two frames? Given frame to frame alignment, our goal is to minimize the number of consecutive frames required for detection, because pixels where motion is undetectable, is proportional to that number (Fig. 5). Another reason for this goal is that residual error after alignment, and the warping process itself, results in significant noise in difference images, which also increases with the number of frames used. The minimum number of frames required for motion estimation is obviously two, the difference of which, however, produces significant ghosting. We use a new approach for detecting ghosts, based on the observation that all blobs obtained using difference of two frames belong to either one of the frames. Specifically, we note that for any blob to be a

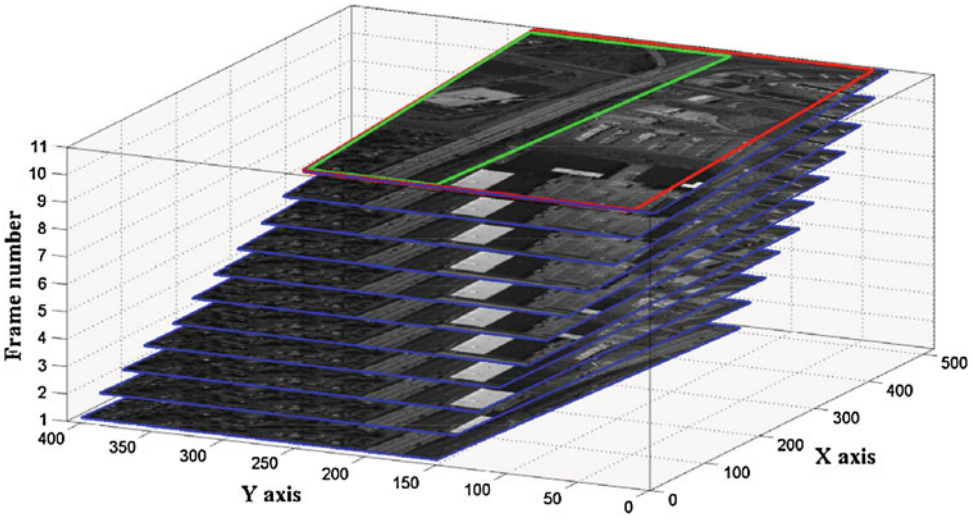


Fig. 5 Difficulties with background learning based detection (Stauffer and Grimson 2000) for fast moving platform. 11 frames aligned to a common reference are shown where each frame's FOV is depicted by *blue polygons*. Using a background model over the first 10 frames will allow detection of objects in only the region inside the *green polygon*, and 54.12 % pixels will not be detectable in the 11th frame. The proposed method uses only two frames, so all pixels in the *red polygon* will be detectable, losing only 4.93 % pixels (Color figure online)

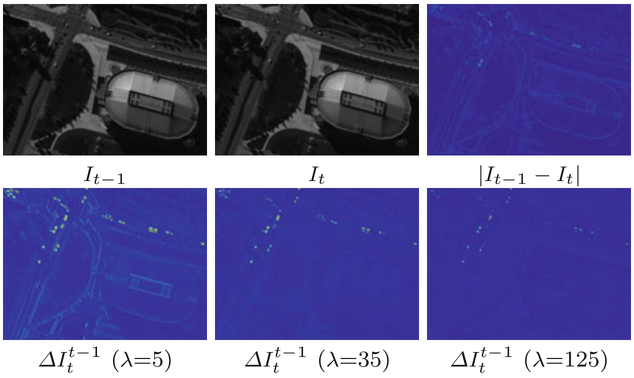


Fig. 6 Background gradient suppression in adaptive two frame difference. Too high a value of λ starts eroding foreground objects. λ = 25 was used in our experiments

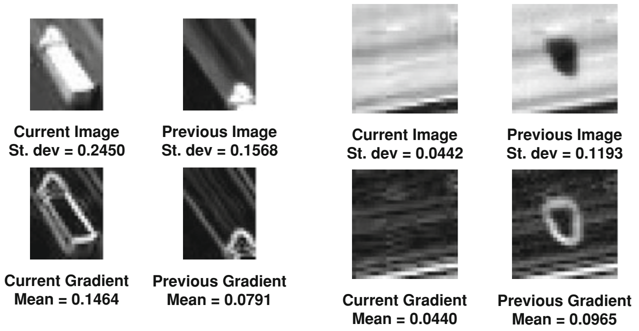


Fig. 7 Ghost disambiguation: blobs from ΔI_t^{t-1} superimposed on previous and current frames. Two examples are shown for bright (*left*) and dark (*right*) objects. Image intensity variance, mean image gradient for true detection are larger than that of the ghost

true moving object, in addition to motion, it must have significantly different intensity relative to the background, or it would not have had a high frame difference.

Adaptive frame difference: The detection method employs the general frame difference function, $F_t = 0$, if $-\gamma < \Delta I_t^{t-1} < \gamma$, and 1 otherwise, where γ, is a positive threshold selected automatically using Otsu's method (Otsu 1979), and

$$\Delta I_t^{t-1} = \frac{I_{t-1} - I_t}{\exp\left(\nabla(I_{t-1} + I_t)\right)^{\lambda}}. \tag{9}$$

$\nabla(.)$ represents the spatial gradient magnitude of the mean of the consecutive images and is used to dilute the frame difference after raising to a power λ (Fig. 6), essentially suppressing some frame difference emanating from regions of high gradients (strong edges in background). The value of λ was fixed at 25 for all our experiments. Both the frame difference, and mean gradient are normalized before computation of Eq. 9.

Ghost disambiguation: Given the set of detected object blobs as $\mathcal{Y}_{t-1:t} = \{\boldsymbol{y}_k\}$, for each blob $\boldsymbol{y}_k$, $k \in [1, Q_{t-1} + Q_t + \xi]$, obtained after connected component analysis of F_t, we wish to obtain two disjoint sets, $\mathcal{Y}_{t-1}$ and $\mathcal{Y}_t$. $\mathcal{Y}_{t-1}$ then becomes the set of blobs where standard deviation of intensity as well as mean of image gradient for previous image pixels belonging to the blob, are larger than that in the current image (Fig. 7). The opposite condition is true for the set of blobs in the current image written as $\mathcal{Y}_t$, and the set of blobs where neither condition is satisfied, i.e., $\mathcal{Y}_\xi$, is discarded as false positives. Some blobs may also be discarded based on size, eccentricity, and mean frame difference within them.

In other words, we know which pixels belong to a detected object, but we don't know which image (previous or current) contributed to the detected blob. For the same set of pixels, we essentially have two blobs in two images, a true object and a ghost. We assume that the background is locally homogenous, so true blobs have a higher inter-pixel intensity difference compared to the ghost blob. Similarly, ghost blobs have a smaller mean gradient (see Fig. 7). This is a reasonable

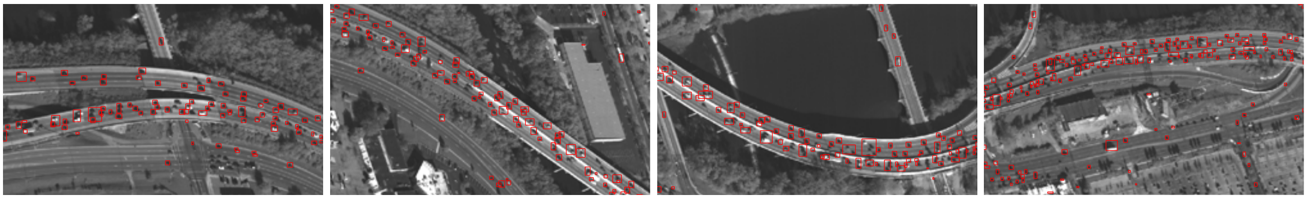


Fig. 8 Object detections in four small disjoint regions of a single camera's FOV are shown as *red rectangles*. Explicit examples of ghost removal can be seen by comparing Fig. 10d and e (Color figure online)

assumption because a moving object could not have been detected even with a background model if its intensity were not different relative to the background. These conditions are applied to actual intensity images, I_{t-1} and I_t, given the blob pixels, rather than on ΔI_t^{t-1}. The detected blob is also dilated by a few pixels, to include part of background as context, for performing the test for ghost disambiguation.

These conditions resemble some of the motion based object detection methods that employ local luminance and contrast variation to isolate moving objects (Cucchiara et al. 2000; Huang et al. 2008; Wang et al. 2008). The main difference in our approach is that we use these local cues for ghost disambiguation, not object detection, which is based on the adaptive two frame difference (Eq. 9). Some examples of detections after ghost removal are shown in Fig. 8. Each measurement $\boldsymbol{y}_k(t)$ is then represented by the vector $\{(\mathbf{p}_k, \mathrm{s}_k, \mathrm{g}_k)\}$, where $\mathbf{p}_k$ is the blob centroid, s_k is the detected area in pixels (object size), and $\mathrm{g}_k = \mu(|\Delta I_t^{t-1}| \in \boldsymbol{y}_k)$ is the mean frame difference. Adaptive frame difference image, ΔI_t^{t-1} is also saved for use during the tracking process.

4.2 Detection Confidence and Object Size

As described before, the small object sizes, variability in shape, and lack of texture and color, may preclude the use of appearance based detection and tracking techniques. The proposed method however, does take into account the consistency in object size, as well as the confidence in detection, which can be expressed as the mean frame difference of the pixels belonging to a blob, i.e., g_k. Given the size s_k and mean frame difference g_k for measurement $\boldsymbol{y}_k(t)$, two more cues can be incorporated into the association likelihood. We therefore include $\mathrm{g}_k(t)$, as the third term in Eq. 8.

Moreover, one of the terms defining object-centric posterior (second term in Eq. 8) incorporates the object size, $\mathrm{s}_k(t)$ in addition to the object motion model. The mean frame difference g_k of the measurement $\boldsymbol{y}_k(t)$ represents the tracker's confidence in observing a measurement, and is especially useful in the case of mis-detections and occlusions, as described in Sect. 4.4. It also significantly reduces the probability of associating a track to a low confidence false positive.

4.3 Candidate Initialization and Propagation

The object-centric likelihood is highly dependent on the object's velocity and acceleration estimates, which are not reliable until after an object has been tracked for at least a few frames. In absence of road orientation, and given low frame rate, greedy nearest neighbor based initialization (zero initial velocity) can be highly erroneous even for very few objects (Fig. 2, top). This ambiguity cannot be resolved unless the direction of motion is explicitly estimated by road detection, etc. For example, (Xiao et al. 2010) proposed the use of GIS which requires platform metadata which may be unreliable or unavailable. Moreover, GIS data itself often needs post-processing and refinement. Heuristics based on coarse road direction estimate (Reilly et al. 2010), and comparisons of spatial layout of objects in consecutive frames (Xiao et al. 2010) have also been used to alleviate this problem, but these incur overhead which is needless given that multiple candidates per object can be maintained to postpone correspondence.

The first few associations for each track are decisive for correct tracking, which is where multiple candidate tracks prove the most important. In the second frame of visibility t, for an object $\boldsymbol{x}_i$, with only a single measurement candidate, $\boldsymbol{x}_i^1(t-1)$, all measurements in the spatial vicinity of the first frame's position are assumed to be feasible candidates regardless of orientation, such that $\forall k \mid \|\mathbf{p}_k(t) - \mathbf{p}_i^1(t-1)\| \leq \mathbf{d}_{\max}$, a new candidate track is spawned for the object $\boldsymbol{x}_i$. $\mathbf{d}_{\max}$ is a fixed distance based on the image resolution and typical object size, and acts as a gating function, and was constant for all experiments. An example of multiple candidate initialization is shown in Fig. 9. The speed of a track candidate can be computed in the second, while the acceleration can be computed starting from the third frame of visibility.

In addition, at any arbitrary frame t, each member in the set of unassociated measurements, is initialized as a new object track with a single candidate, each with a single track point, and probability $\mathrm{g}_k(t)$. Furthermore, all associated measurements, whose un-normalized maximum association likelihood (among all tracks) is less than half the mean of maximum association likelihoods (among all candidates of all tracks), are also initialized as new tracks, i.e., handling of the

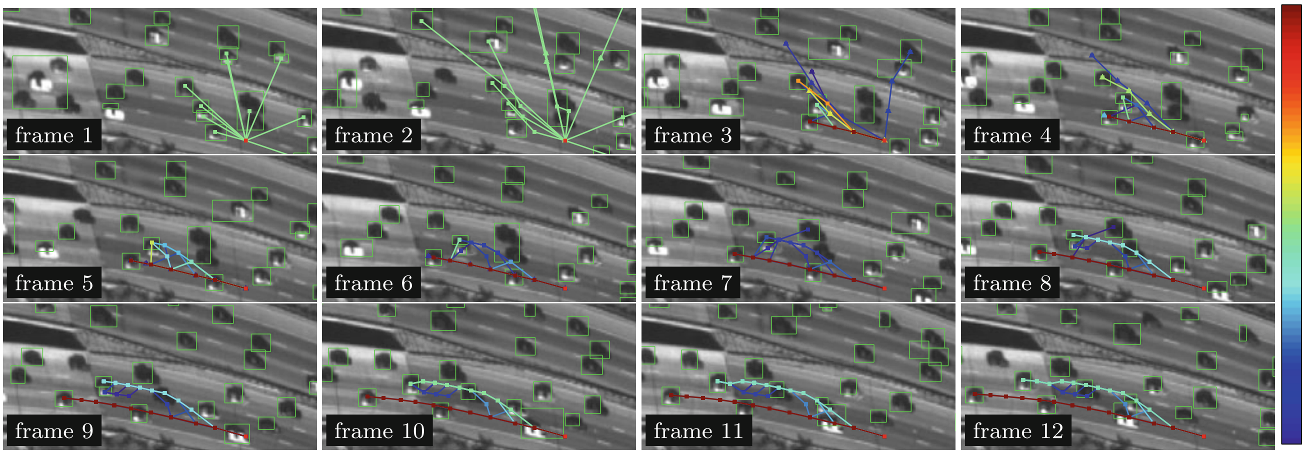


Fig. 9 Evolution of candidates of a single target, over 12 frames. Each *image* shows top 10 candidates for the object, where *filled square* represent actual, and *filled triangle* depict hypothetical measurements. *Color* of each candidate track depicts relative probability as per the *color bar*. Detections in each frame are shown as *green bounding boxes*. Notice that all candidates are equally likely up to second frame, due to lack of acceleration estimate, but the correct candidate soon emerges as most likely (Color figure online)

case where a new object may appear (enter) near an existing track's predicted location.

4.4 Occlusion and Mis-detection Handling

Since objects in high altitude aerial sequences are frequently mis-detected because of illumination changes (due to changes in sensor gain) and shadows, and undergo occlusions in urban traffic scenarios (e.g., trees or bridges, etc.), it may sometimes be impossible to find any reasonable association for the best and correct hypothesis. Furthermore, in absence of road network knowledge or data driven learning, it is nearly impossible to discern between a valid occlusion like bridges or trees, and a mis-detection like a dark object in a shadow or object with appearance similar to background. In this case, the proposed algorithm adds a hypothetical measurement, $\hat{\mathbf{y}}_k$, where $k > Q_t$, to the ensemble of possible candidates, anticipating that the object will soon come out of occlusion. This process however carries an implicit penalty arising from the association likelihood representing mean frame difference, $\mathrm{g}_k(t)$. Specifically, the hypothetical measurement vector for a candidate $\boldsymbol{x}_i^j(t-1)$, is defined as,

$$\hat{\mathbf{y}}_k(t) = \Big[\mathbf{p}_i^j(t-1) + \mathbf{v}_i^j(t-1) + \alpha_i^j(t-1),\quad s_i^j(t-1),\quad \mu(|\Delta I_t^{t-1}| \in \hat{\mathbf{y}}_k)\Big], \tag{10}$$

that is, a hypothetical detection is assumed to have been observed at the predicted location, with a size equal to the object's size in the last frame, and a confidence, $\mathrm{g}_k(t)$, equal to the mean gray area corresponding to an area in the current difference image, which is the same size as the object's previously observed size. This confidence computation is the reason that the adaptive frame difference image ΔI_t^{t-1} is stored.

If the hypothetical measurement is indeed a mis-detection, this penalty would not be too severe, which is in contrast to existing approaches that rely on fixed weight occlusion nodes (Reilly et al. 2010). A hypothetical detection is especially useful in the case of mis-detection due to high detection threshold. In case a track has actually ended, the process of hypothetical measurement insertion is not continued for more than a fixed number of frames. A hypothetical measurement is also useful in case of blob merging, where association to centroid of merged detection will likely result in change of direction as opposed to that of hypothetical detection. An example of such a scenario is shown in Fig. 10. The process of hypothetical measurement introduction essentially converts the space of observations (measurements/detections) from discrete ($\mathcal{Y}_t$), to continuous, such that the probability of an arbitrary observation at any pixel, with any size can be computed. The observation distribution is analogous to a correlation surface or classifier output surface for motion based detections.

Given the discussion in Sect. 4, the independent component P_i of the probability for each candidate (Eq. 3) can now be computed. Furthermore, a structured method for generation of new object tracks, as well as for spawning new candidate tracks for existing objects has been described. In Sect. 5, we describe methods to compute the context dependent components, P_f and P_m.

5 Context Aware Association

One of the main goals of the proposed work is to avoid hard assignments, as well as the 1–1 correspondence constraint.

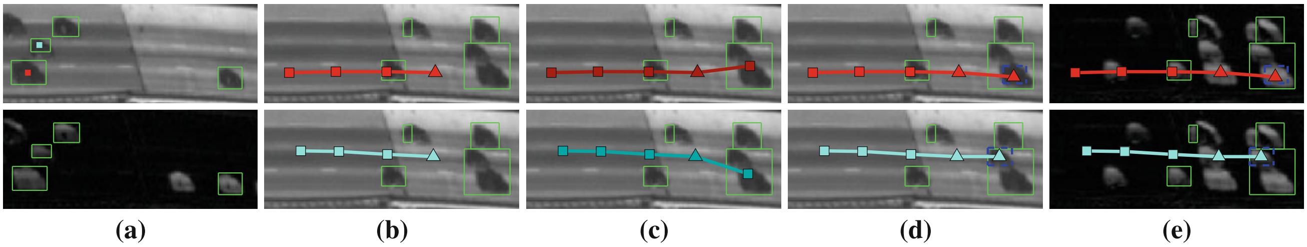


Fig. 10 Effect of *Weighted* hypothetical measurement (*filled triangle* instead of *filled square*) on detection merging. Tracks of two objects travelling *left* to *right* in *parallel* are shown in *red* and *cyan*. Detections in the frame are shown as *green boxes*. **a** shows the initialization of each of the two tracks in frame 318 of sequence 2, overlaid on I_{318} on *top*, and the difference image ΔI_{318}^{317} on *bottom*. **b** shows the best candidate for each track at frame 321, overlaid on I_{322}. **c** shows the second best candidate of each track in frame 322, each of which corresponds to association with the merged blob center, while **d** displays the best candidates of each that is created by association to a weighted hypothetical measurement, depicted by *blue dotted boxes*. The images in **e** contain the corresponding difference image, where the hypothetical measurements obviously have non-zero mean *gray* area, i.e., $g_k(t)$. Notice that the ghosts in **e** are not detected as objects (Color figure online)

The relaxation of this constraint however, introduces the problem of track merging. In absence of matching functions that discourage measurement waste, it is possible that a track switching its label incorrectly, will result in duplicate tracks for one object, and none for the other. Even if the duplicate is not chosen as the best candidate, it may serve to help discard the correct one based on a lower probability. The lost object's subsequent measurements will not only be wasted, but there will be no reacquisition. Only initiation of a new object track will take place, because unassociated detections spawn new tracks, essentially resulting in broken tracks.

In the subsequent subsections, we propose two novel data association constraints which are also applicable to conventional MTT techniques. The goal of these techniques is to consider the effects that neighboring objects have on tracking of a particular target, while evaluating the association likelihood for that target. They also reduce measurement waste, that can result from track merging due to detection sharing.

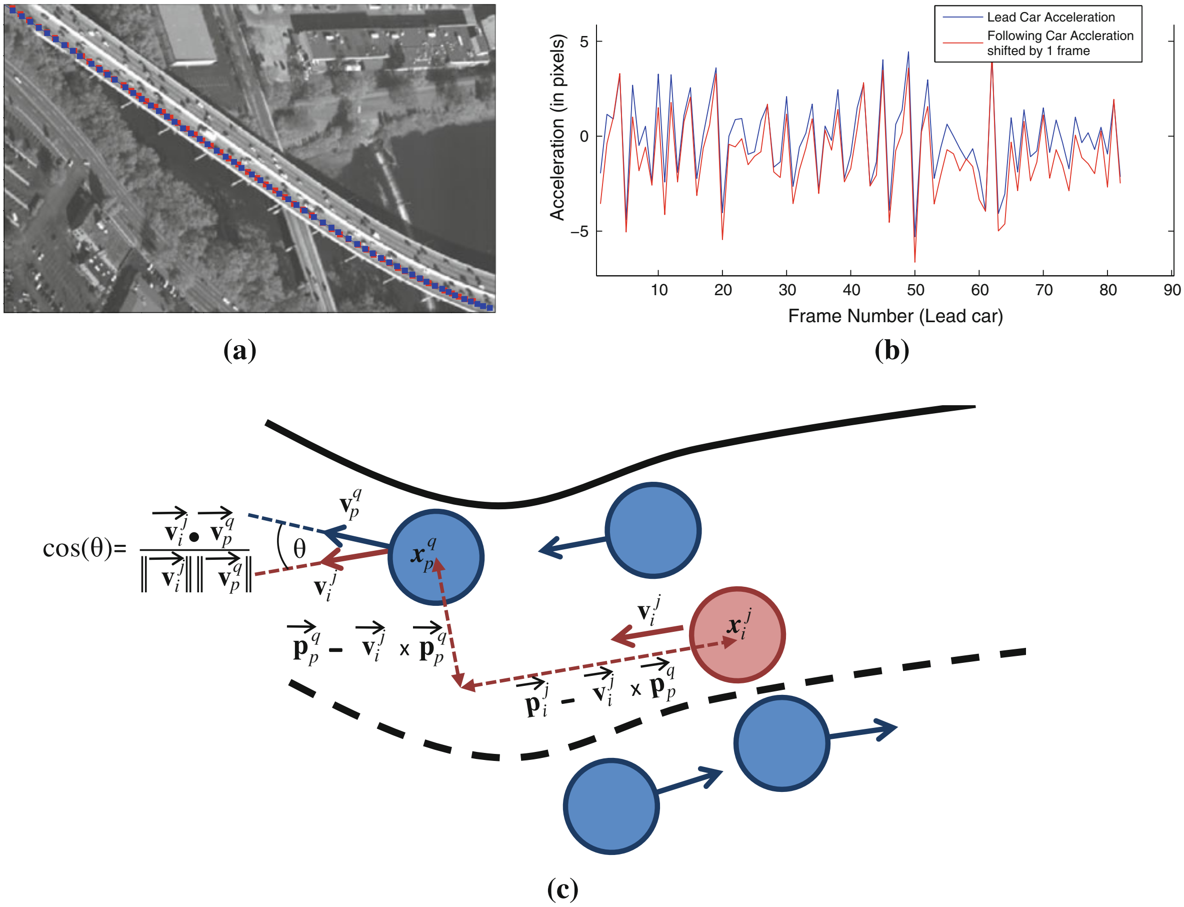
5.1 Vehicle Following Model

The correlation between the accelerations of vehicles following one another has been established by research in transportation theory and traffic analysis. Various models have been proposed in these fields, which attempt to mathematically explain how drivers tend to follow one another in a stream of traffic. These models include Fuzzy logic (Kikuchi and Chakroborty 1992), cellular automata (Nagel and Schreckenberg 1992), differential and difference equations (Newell 1961), and are useful in planning and analysis of transportation systems. However, these methods have not been used in the area of target tracking in dense urban traffic scenarios, possibly due of lack of visual data that challenges conventional methods. The data set under consideration presents precisely such scenarios where small object sizes, low frame rates, and the sheer number of targets preclude many traditional cues like proximity, maximum or constant velocity, rigidity, or locally similar motion. We observe in the data set that for free flowing traffic, the instantaneous accelerations of leading-following car pairs are closely related, but shifted in time, as illustrated in Fig. 11a, b. For the purpose of evaluating the effect of the lead vehicle's motion on the following vehicle, we employ the classical stimulus response model, the Gazis–Herman–Rothery (GHR) model (Gazis et al. 1959), as generalized by Edie (1960):

$$\frac{d^2\mathbf{p}_F(t)}{dt^2} = \frac{\rho\left[\frac{d\mathbf{p}_F(t-\Delta t)}{dt}\right]^m}{\left[\mathbf{p}_L(t-\Delta t)-\mathbf{p}_F(t-\Delta t)\right]^l} \cdot \left[\frac{d\mathbf{p}_L(t-\Delta t)}{dt} - \frac{d\mathbf{p}_F(t-\Delta t)}{dt}\right], \tag{11}$$

where $\mathbf{p}_L$ and $\mathbf{p}_F$ are the 2d locations of the lead and following vehicles respectively, ρ is the sensitivity coefficient, a high value of which indicates high response intensity for the following driver, Δt is the response time, while m and l are two parameters, which have empirically been found to be between $m = 0$–2, and $l = 1$–2 for uncongested traffic. The above can be described intuitively by noticing that the predicted acceleration of an object is, directly proportional to its current velocity, directly proportional to the difference between its velocity and that of the lead vehicle, and inversely proportional to the distance between the two.

Let the random variable $F_{i,j}^{p,q}$ denote the event that the object represented by track candidate $\boldsymbol{x}_i^j(t-1)$ is *following* the one represented by track candidate $\boldsymbol{x}_p^q(t-1)$. We then write the probability P_f of observing a particular state $\boldsymbol{x}_i^j(t)$ of the ith object, given it is following other objects in its immediate neighborhood $\mathcal{X}_i'(t)$, as the joint probability of, (a) observing similar accelerations from a particular association and the car following model, and (b) the event that object under consideration is part of a lead-follow pair:


Lead Car Acceleration
Following Car Accleration shifted by 1 frame
Acceleration (in pixels)
5
0
−5
10
20
30
40
50
60
70
80
90
Frame Number (Lead car)
cos(θ)=
$\vec{\mathbf{v}}_i^j \bullet \vec{\mathbf{v}}_p^q$
$\|\vec{\mathbf{v}}_i^j\|\|\vec{\mathbf{v}}_p^q\|$
θ
$\mathbf{v}_p^q$
$\mathbf{v}_i^j$
$\boldsymbol{x}_p^q$
$\boldsymbol{x}_i^j$
$\vec{\mathbf{p}}_p^q - \vec{\mathbf{v}}_i^j \times \vec{\mathbf{p}}_p^q$
$\vec{\mathbf{p}}_i^j - \vec{\mathbf{v}}_i^j \times \vec{\mathbf{p}}_p^q$

(a) **(b)**

(c)

Fig. 11 Modeling vehicle following behavior: **a** tracks of a lead-following vehicle pair in *blue* and *red* resp., **b** instantaneous accelerations plot of lead and following vehicles in *blue* and *red* resp. Accelerations of following vehicle are remarkably similar to that of the lead vehicle when shifted left by 1 frame, **c** illustrates the process of lead vehicle probability computation for all objects near a vehicle, and depends on similarity of direction (θ), and components of vector between object locations that are parallel and normal to the velocity vector. Probability for oppositely moving vehicles is almost 0 (since, $\cos(\theta) \approx -1$) (Color figure online)

$$P_f\left(\boldsymbol{x}_i^j(t)\middle|\mathcal{X}_i'(t),\mathcal{Y}_t\right) = P_f\left(\boldsymbol{x}_i^j(t)\middle|\mathcal{X}_i'(t)\right) = \sum_{p\in[1,T_t]-\{i\}}\ \sum_{q\in[1,N_p(t)]} P\left(\boldsymbol{x}_i^j(t),\, F_{i,j}^{p,q}\right), \qquad (12)$$

where the right hand side can be decomposed using Bayes rule as,

$$P\left(\boldsymbol{x}_i^j(t),\, F_{i,j}^{p,q}\right) = P\left(\boldsymbol{x}_i^j(t)\middle|F_{i,j}^{p,q}\right)\cdot P\left(F_{i,j}^{p,q}\right), \qquad (13)$$

which can be thought of as a weighted mean of, the probabilities of different accelerations predicted by the car following model (first term), where the weight is given by the probability that the two cars are indeed a lead-follow pair (second term). The first term can easily be computed using the model given by Eq. 11 and using the mean and covariance of the object state's acceleration. Assuming the parameters in Eq. 11 to have values as $\Delta t = 1$ frame (0.5 s), $m = 1.5$, $l = 1$, and $\rho = 0.25$, and using previous notation, we compute the conditional likelihood for the jth candidate of the ith object $\boldsymbol{x}_i^j$, given it is following candidate $\boldsymbol{x}_p^q$ as,

$$P\left(\boldsymbol{x}_i^j(t)\middle|F_{i,j}^{p,q}\right) = \mathcal{N}\left(\rho\left[\mathbf{v}_i^j(t-1)\right]^m \cdot \frac{\left[\mathbf{v}_p^q(t-1) - \mathbf{v}_i^j(t-1)\right]}{\|\mathbf{p}_p^q(t-1) - \mathbf{p}_i^j(t-1)\|^l};\, \alpha_i^j(t),\, \Sigma_{i,j}(t)\right). \qquad (14)$$

The idea again, is to evaluate probability of the acceleration obtained by vehicle following model based prediction, given how the object's acceleration (from data-association) is distributed.

Finally, the probability $P\left(F_{i,j}^{p,q}\right)$, of object $\boldsymbol{x}_p^q$, being the lead vehicle of $\boldsymbol{x}_i^j$, is computed by simply shooting a ray, from the current location of $\boldsymbol{x}_i^j$, in the direction of current velocity $\mathbf{v}_i^j$, and finding the perpendicular distance of $\mathbf{x}_p^q$ from the ray, and distance of point of intersection from $\mathbf{x}_i^j$, where the ray obviously is the current velocity vector, $\mathbf{v}_i^j$ (see Fig. 11).

Additionally, the similarity between the current directions of motion of the two objects is also computed, so as to disregard influence from nearby but oppositely moving vehicles. We can therefore write,

$$P\left(F_{i,j}^{p,q}\right) = \frac{1}{2}\left(\frac{\overrightarrow{\mathbf{v}}_i^j \bullet \overrightarrow{\mathbf{v}}_p^q}{\|\overrightarrow{\mathbf{v}}_i^j\|\|\overrightarrow{\mathbf{v}}_p^q\|}+1\right) \cdot exp\Big\{-\big(\|\overrightarrow{\mathbf{p}}_i^j - \overrightarrow{\mathbf{v}}_i^j \times \overrightarrow{\mathbf{p}}_p^q\| + \|\overrightarrow{\mathbf{p}}_p^q - \overrightarrow{\mathbf{v}}_i^j \times \overrightarrow{\mathbf{p}}_p^q\|\big)\Big\}, \tag{15}$$

where $\overrightarrow{\mathbf{p}}$ and $\overrightarrow{\mathbf{v}}$ represent position and velocity vectors in homogenous coordinates, and '$\bullet$' and '$\times$' are the dot and cross products respectively. In other words, the probability represents the similarity between directions of vectors $\mathbf{v}_i^j$ and $\mathbf{v}_p^q$, coupled with the distance between their tails. To avoid useless computations, in our experiments, $P\left(F_{i,j}^{p,q}\right)$ was simply set to 0 using a gating function, wherever the distance between $\mathbf{x}_i^j$ and $\mathbf{x}_p^q$ is greater than $3\|\mathbf{v}_i^j\|$. Furthermore, it is made sure that the sum of probabilities $P\left(F_{i,j}^{p,q}\right)$ for all possible lead vehicles of $\mathbf{x}_i^j$ is 1, that is, $\sum_{p\in[1,T_t]-\{i\}}\sum_{q\in[1,N_p(t)]} P\left(F_{i,j}^{p,q}\right) = 1$. This normalization is omitted in Eq. 15 for clarity. Equations 13, 14, and 15 can now be plugged into Eq. 12 to obtain probability P_f of a candidate track $\boldsymbol{x}_i^j(t)$ according to the vehicle following model, which can be thought of as an intelligent motion similarity constraint for proximal targets. It should be noticed that the proposed vehicle following model does not actually choose a *single* lead vehicle, which is rather difficult for high density traffic in absence of road orientation estimates and lane information. Instead, all objects in the vicinity are assumed to be the lead vehicles and the corresponding probabilities are evaluated. The vehicles moving in opposite direction obviously do not significantly affect the estimate.

5.2 Avoidance of Track Intersection

Another important cue in MTT from nadir views in structured scenes is that correct tracks do not intersect, at least not within a small temporal window (Fig. 12). This cue is often used as a post-processing step to remove tracks that intersect with other ones (Xiao et al. 2010). However, the only option at that stage is to either discard detections within the removed track, or assign it to be under occlusion. One of the novel contributions of the proposed algorithm is to bring this post-processing cue into the association likelihood computation, thereby discouraging track intersection in general, but allowing it in absence of other viable alternatives. Furthermore, the relaxation of 1–1 correspondence constraint may cause problems, e.g., the best candidates for two nearby tracks may share measurements, while ignoring other valid measurements. The proposed idea implicitly alleviates this problem as well, because candidates merging onto the same measurement are penalized, as they intersect *at* the measurement. Similarly, measurement wastage is discouraged because measurements associated with fewer candidates have a smaller penalty compared to ones associated with more candidates.

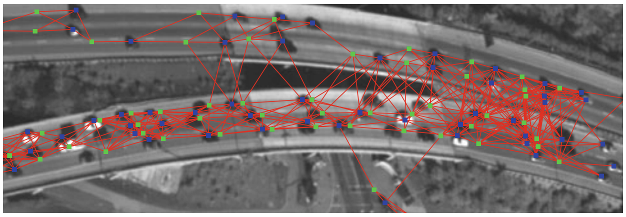


Fig. 12 All possible associations for all candidates of a few objects in consecutive frames. Track locations in previous frame and observations in current frame are shown by *green square* and *blue square* resp. Majority of these are infeasible due to intersection with other associations (Color figure online)

Given the sets of candidates for all objects, the problem of intersection detection can be defined as an exhaustive test of intersection between pairs of line segments, such that the two ends of every line segment correspond to measurements in distinct consecutive frames. Due to the extremely large number of measurements in each frame, and the fact that the number of line segments can possibly be quadratic in the number of measurements, the simple approach would require $O(N^2)$ time for N line segments, where N can be $Q_{t-1} \cdot Q_t$ in the worst case. We employ the efficient Shamos-Hoey sweep line algorithm (Shamos and Hoey 1976) for segment intersection testing which runs in $O(NlogN)$ time. Notice that we do not require the actual points of intersection, only a binary result for each segment pair indicating whether an intersection is present. Let us denote such a result as S, such that $S(i, j, p, q)$ is 1, if the two candidates $\boldsymbol{x}_i^j(t)$ and $\boldsymbol{x}_p^q(t)$ intersect in the past two frames, and 0 otherwise, where $1 \le i, p \le T_t, i \ne p, 1 \le j \le N_i(t)$, and $1 \le q \le N_p(t)$. Moreover, track candidates that carry occlusion nodes in the past few frames are exempt from this test ($S = 0$ for these), since some of them represent vehicles moving under bridges, etc., and therefore in directions perpendicular to normal traffic, resulting in valid intersections. We now define a function L, such that,

$$L\left(\boldsymbol{x}_i^j(t), \boldsymbol{x}_p^q(t)\right) = \frac{P\left(\boldsymbol{x}_i^j(t)|\mathcal{Y}_{1:t}\right)}{P\left(\boldsymbol{x}_p^q(t)|\mathcal{Y}_{1:t}\right)^{S(i,j,p,q)}}, \tag{16}$$

where $L \in [P\left(\boldsymbol{x}_i^j(t)|\mathcal{Y}_{1:t}\right), \infty]$. Therefore, the value of L for two candidate tracks is equal to the probability of the first if they do not intersect; it is equal to 1 if they intersect and are equally likely; it tends to the probability of the first as that of the second reaches 1; and it tends to infinity as the probability of the second reaches 0. We use this function to define the following:

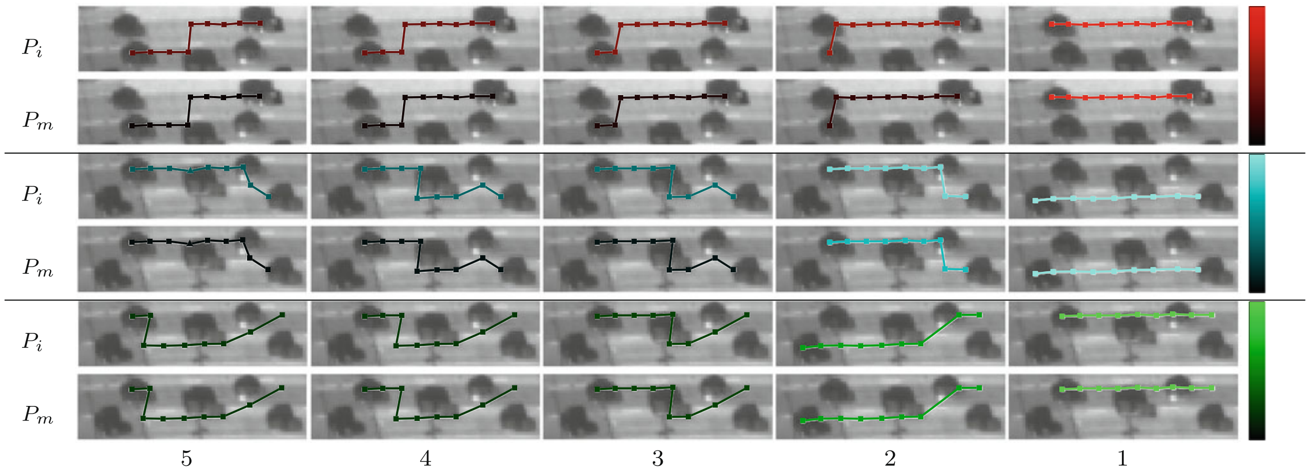


Fig. 13 Effect of track intersection discouragement: Top five candidates for three vehicles travelling *right* to *left* in parallel lanes. The *two rows* for each track correspond to probabilities P_i and P_m shown by shades of *red*, *cyan*, and *green* (brighter meaning high probability as per *colorbars*). Notice that the disparity among candidate probabilities is much higher for P_m than P_i. Track intersection is minimized by choosing candidate 1 for each track (Color figure online)

$$P_m\left(\boldsymbol{x}_i^j(t)\middle|\mathcal{X}_i'(t),\mathcal{Y}_t\right) = \frac{\sum\limits_{p\in[1,T_t]-\{i\}}\ \sum\limits_{q\in[1,N_p(t)]} L\left(\boldsymbol{x}_i^j(t),\boldsymbol{x}_p^q(t)\right)}{\sum\limits_{j'\in[1,N_i(t)]}\ \sum\limits_{a\in[1,T_t]-\{i\}}\ \sum\limits_{b\in[1,N_a(t)]} L\left(\boldsymbol{x}_i^{j'}(t),\boldsymbol{x}_a^b(t)\right)}. \tag{17}$$

In other words, if two candidates intersect within last two frames, the *intersectee* candidate's likelihood changes by a factor of the *intersector* candidate's likelihood, and vice versa. Consequently, the candidate that intersects or shares detections with the least number of tracks, has the highest probability according to this cue, where the increased probability is a function of its own and the intersecting tracks' probabilities. It should be noticed that dependence of P_m on $\mathcal{Y}_t$ is a manifestation of using the posterior, $P\left(\boldsymbol{x}_i^j(t)|\mathcal{Y}_{1:t}\right)$, in the function L. Notice also that evaluation of P_m requires computation of *all* object-centric likelihoods for the *current* frame, and is therefore part of a second loop over the tracks. An effect of this important cue can be seen in Fig. 13, which alone is helpful in the following ways:

- Mitigating negative consequences of allowing detection sharing between objects,
- Discouraging overzealous merging of track candidates (and eventually tracks) onto a single measurement,
- Implicitly imposing a local velocity similarity constraint by penalizing intersecting candidates,
- Encouraging candidates with hypothetical measurements in case of measurement merging (i.e., merged blobs), by penalizing intersection, i.e., distinct track candidates associating with the same measurement (see Fig. 10 for an example of this scenario) and,
- *Allowing* intersection and merging of tracks in absence of better options, instead of treating them as impossible.

It should however be noticed that the convergence of two distinct tracks onto a single 'correct' detection will almost never result in the best candidate. An example of such a scenario can be seen in Fig. 10, where it makes sense to allow detection sharing, since it is indeed a merged detection, but it is unlikely that a hypothetical detection will get a lower overall probability, given higher values from motion model, vehicle following, as well as track intersection cues. If on the other hand the hypothetical detection has low mean gray area (e.g., due to permanent occlusion like an overpass), the eventual probability after insertion of a few hypothetical detections, and subsequent actual ones, will be higher for the correct candidate due to better adherence to the motion model and intersection avoidance.

Even though it is rare, the merging of two distinct object trajectories to a single detection is still a possibility and within the tracking process, there is no constraint to prevent it (i.e., 1–1 correspondence). If such a case arises, the final decision (Eq. 2) favors the optimal candidate of the track with the higher likelihood or lower overall cost. For the other track, the next best candidate is selected as the optimal one. The scenario under consideration, i.e., two tracks merging for one or more detections, is detected by computing the number of *non-hypothetical* detections (e.g., the merged detection in Fig. 10) that are shared between the two tracks, normalized with respect to the respective track lengths. The tracks are considered to be merged if more than 40 % of the

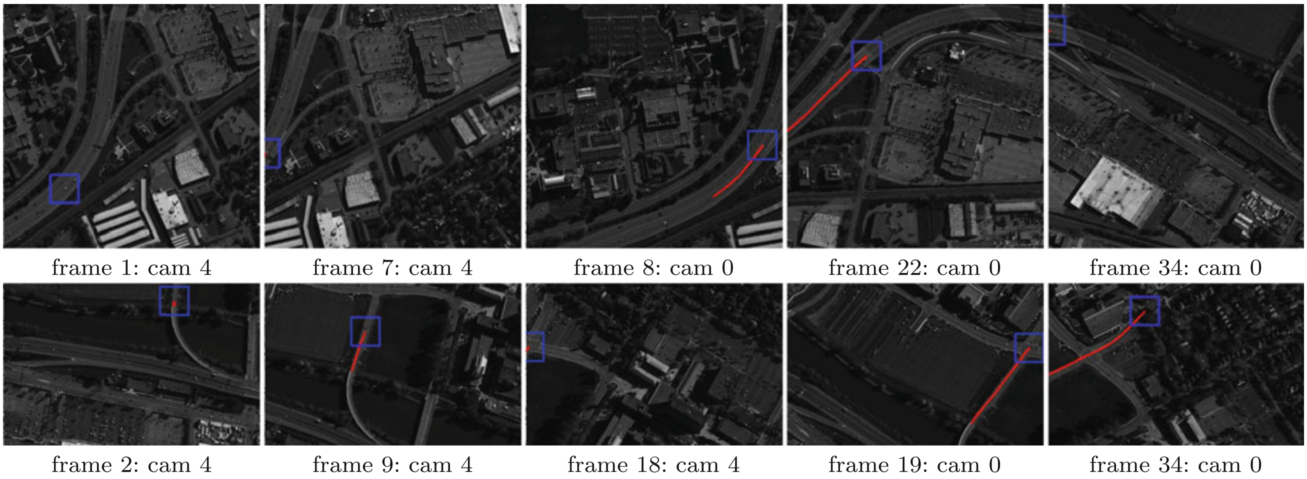


Fig. 14 Object tracking across cameras for two objects shown in each *row*. Trajectory is shown as a *red line*, and object's current location as a *blue rectangle* (Color figure online)

non-hypothetical detections of each are identical. A few quantitative remarks about the frequency of occurrence of this scenario are made in Sect. 7.

6 Multiple Camera Tracking

Wide area aerial data often benefits from simultaneous image capture by multiple cameras, as is the case in the CLIF data (USAF 2006), where six high resolution cameras capture data at every frame. Although the inter-camera transformations are easily computable due to reasonable overlap in their FOVs, and remain fixed over time, multiple camera tracking is still challenging due to a number of reasons. First, the size of the resultant mosaic image at every frame (>70 million pixels) itself prohibits sophisticated object detection algorithms. Moreover, the difference in camera gains and contrast imply the requirement of gain adjustment or histogram equalization preprocessing before background model learning or frame differencing. Second, even the minor residual errors in transformation computation and image warping steps are significant enough to throw off the tracker, e.g., detections expected on the road may now appear in the opposite lane of traffic, etc. Although the tracking algorithm's multiframe association probability computation mitigates the latter problem to some extent, the former still requires computationally expensive preprocessing, e.g., Reilly et al. (2010) performed gain and brightness equalization across cameras to stitch a mosaic before performing object detection at the mosaic level.

Our proposed framework employs a simpler approach, possible due to the inherent detection sharing capability. Instead of preprocessing and camera to camera image warping before detection, objects are first detected in each camera using the proposed two-frame difference approach. The detections obtained in all camera frames at a particular time, are then transformed to a single camera's reference at that frame, bypassing camera-camera image warping. The resultant set of detections now contain duplicate detections in the regions visible in multiple cameras' fields of view. This drawback however poses no significant limitations on the tracking performance. In the rare case that the duplicate detections of a particular object spawn multiple candidate tracks, one of them will attain a higher probability, even if marginally so. Two examples of camera handover during tracking are shown in Fig. 14.

7 Experiments and Results

The proposed method has been tested on the challenging CLIF dataset (USAF 2006). CLIF stands for Columbus Large Image Format, and consists of sequences captured from a UAV around the OSU campus. The sequences are captured from an altitude of about 7,000 ft (2.1 km) at a rate of 2 frames per second. The sensor consists of six cameras arranged in a 2 × 3 array (see Fig. 1, top). The typical resolution per frame per camera is 4,008 × 2,672 but the high platform altitude results in very few pixels on target. Also notice that all CLIF imagery is grayscale, making appearance cues much less discriminative. The candidates for each track are pruned to ten candidates ($N_i \leq 10$), but only after track length is at least 5. No pruning is performed for shorter, new tracks so the acceleration estimates are established. All experiments reported were ran using Matlab implementation on a quad core machine. In terms of speed, the tracking part of the proposed approach ran 7.6483 s per frame, or 0.13 frames per second. Using the same set of detections, bipartite graph matching (using (Munkres 1957) took an average of about 57 s per frame. There is potential to optimize the algorithm for improved speed but our approach is significantly faster than simple linear assignment based association. Similarly the storage requirements of our algorithm were also reasonable at 20.4986 MB per frame. Sequence specific numbers for run times and memory requirements are reported in Table 1.

The frames in the data set were divided into several sequences by selecting frames such that a high percentage of visible objects persist in the camera view for a maximum number of frames, so that the tracking results are meaningful. This is because the UAV platform has a much greater speed than the speeds of objects, and it is difficult to generalize performance if objects are not visible for more than a few frames. Detailed specifications of the sequences are listed later along with performance analysis for each sequence.

The idea behind the choice of frames for each of the four sequences is that the frames chosen should have at least one major road or highway, in addition to obviously a large number of smaller streets, parking lots, and intersections, etc. This is important for two main reasons:

(1) Quite a few novelties in terms of data association, object detection, as well as specific steps geared towards bounding the computation and memory requirements, come into play only when there is a large number of point correspondences to be established. If the sequences had fewer objects, some of the existing techniques may have been able to perform adequately. It should be noticed however, that the chosen sequences are by no means devoid of low density traffic, or singular targets moving without other objects in context.

(2) The high platform speed implies that objects do not persist in the field of view for more than a couple of tens of frames. The choice of frames with high density traffic at least allows us to test the proposed approach for tracking a large number of objects within a few frames. Objects with few or no spatially proximal confuser objects are obviously easier to track, and given that they will be visible for only 10–20 frames, the problem would be much easier for regions of low traffic density.

Some explanatory numbers for one of the four sequences (sequence # 3) are computed for additional insight and reported in Fig. 15. Figure 15a shows the very large number of detections per camera for each frame in the sequence. An obvious observation in this plot is the abruptness with which the number of detections change over time. This is due to two main reasons. First, the sequence consists of frames where the airborne camera traverses a busy highway resulting in a sharp increase and subsequent decrease in the number of objects detected. More importantly however, the fixed threshold γ applied to the adaptive frame difference (Eq. 9), is sometimes not optimal due to the difference in camera gain in consecutive frames. This is precisely the reason that discrete, thresholded, binary detections, $\mathcal{Y}_t$, are primarily used for initialization only, and the hypothetical detections cater for the false negative detections. False positives are responsible for track initializations but such tracks cannot continue for more than a couple of frames due to multiple reasons. First, the false positives are often very intermittent and get low confidence for subsequent corresponding hypothetical detections. Second, even for relatively persistent false positives, such as regions on the edges of roads, a low confidence is obtained for vehicle following due to inconsistency with respect to motion of spatially proximal objects. False positives seldom affect targets that have been correctly initialized because over a few frames after a candidate is associate with false detection, the process needs to be discontinued for that candidate. Multiple examples of this scenario can be seen in Fig. 16.

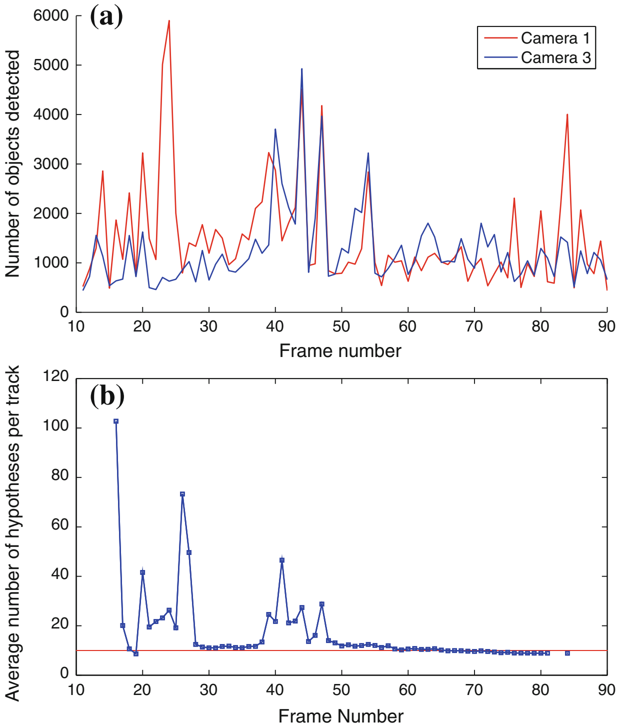


Fig. 15 Explanatory quantitative figures computed for sequence 3: **a** number of objects detected in each camera for sequence 3, **b** average number of viable candidates per track at each frame of sequence 3. The *red line* is drawn at 10, which is the maximum number of propagated candidates after pruning (Color figure online)

Figure 15b shows the corresponding average number of candidates per track being retained at each frame. First, notice that the number of candidates for frames 10–15 are not plotted. The reason is that these are potentially large number of candidates during a time when the candidate pruning process has not started yet. As mentioned earlier, candidates are not discarded until the track is at least 5 frames long. The second

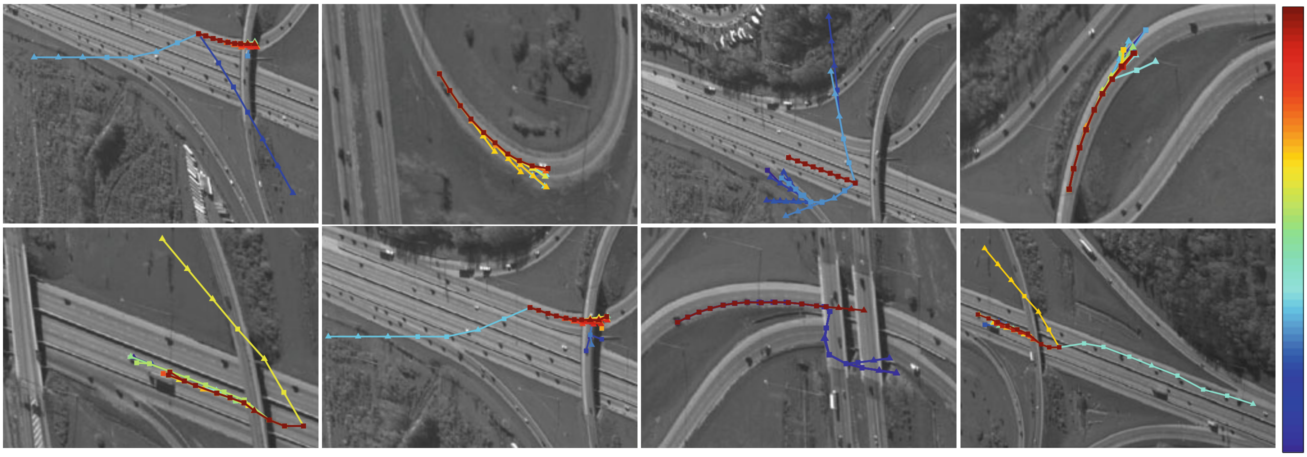


Fig. 16 Top 10 candidates for eight tracks. The relative probability of each is represented by its *color* according to the *color bar*. Observed detections are shown as *filled square*, and hypothetical as *filled triangle*. Notice that the correct candidate usually has the highest likelihood (*dark red*), and occlusions are handled implicitly by incorporation of hypothetical detections, *filled triangle* (Color figure online)

point to note in this plot is that despite the very large number of objects, aggressive pruning makes maintenance of per object candidates possible, and keeps the computational and storage requirements under control.

Quantifying false positive detections As listed in Table 1, only a few (30–100) objects have been manually groundtruthed in each of the sequences, while it can be observed that each frame of video contains thousands of objects. Computation of mis-detections is possible if only the ground-truthed objects are taken into account. The detection rate obviously considers the number of objects that were present but could not be detected. False detection rate on the other hand, requires manual counting of each object that was falsely detected. Given the large number of true objects (and detections), computation of such statistics for all video sequences is a prohibitively laborious task.

We have however performed such manual labeling for a small region (918 × 655 - 1/18th of a frame) in a single frame for one of the four video sequences, as shown in Fig. 17. It can be easily observed that performing such quantitative evaluation for all the four sequences will be a daunting task. Manually performed quantitative evaluation indicates that a total of 110 moving vehicles have been detected correctly as single objects (true positives), while 18 have been mis-detected (false negatives). Despite the large background regions with strong gradients, there are only 8 false positives. There were 7 additional detections which are correct, but contain multiple vehicles within the bounding box. This however is not a significant problem in the proposed approach (see Fig. 10 for an example of handling of merged detections).

In any case, the influence of hard detection thresholds in the proposed approach is very minimal, due to the use of state and observation joint probability in the final likelihood score, instead of the traditional conditional as mentioned in the paper. In other words, the actual detections are used solely for track initialization. False positives can be greatly reduced by keeping a high detection threshold, while the resulting mis-detections are easily mitigated by using the detection confidence, i.e., the mean gray area. That is the reason why the proposed framework emphasizes computation of a clean, crisp, and adaptively filtered two-frame difference. Instead of relying on sophisticated but possibly unreliable thresholds to strictly divide the observed imagery into background and foreground regions, the idea is to instead rely directly on the observed difference, which is diluted using the image gradients to reduce effects of alignment residue and noise. It can also be observed that although we could achieve

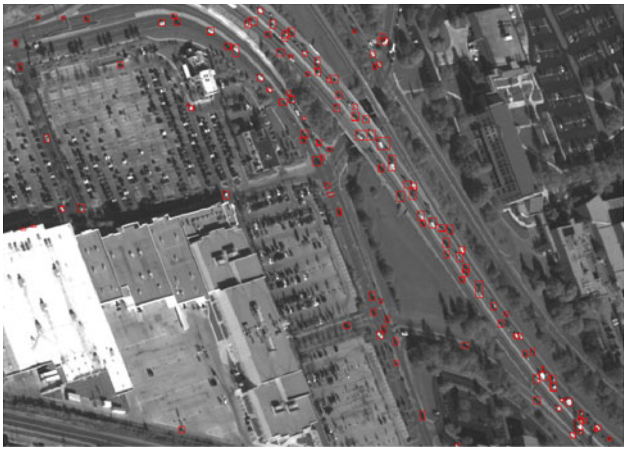


Fig. 17 Quantifying object detection performance in a small 918 × 655, region of frame 318 of the CLIF dataset. Detections are shown as *red bounding boxes*. Performance numbers are reported and explained in text (Color figure online)

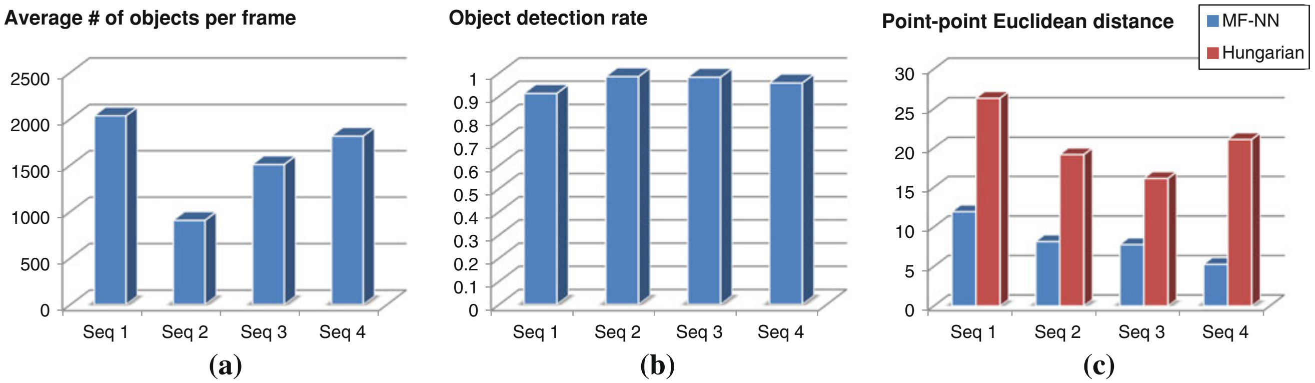


Fig. 18 Quantitative results obtained for four sequences: **a** average number of objects detected per frame; **b** object detection rate considering the ground truthed objects; and **c** average point to point Euclidean distance between ground truth tracks and tracks obtained using bipartite graph matching and the proposed method. Actual numbers are reported in Table 1

perfect ODR by decreasing detection threshold, the resulting increase in the number of false positives will adversely affect tracking metrics including fragmentation and completeness. These quantitative measures of tracking are therefore indirect indication of the low number and minimal effect of false positive detections.

Quantitative comparison We compared the performance of our approach with a conventional multiple target tracker, which optimizes greedy nearest neighbor search using bipartite graph matching. The linear assignment problem is solved using the Hungarian algorithm (Munkres 1957). The baseline tracker uses constant velocity model, where velocities are initialized to be 0 (therefore simple nearest neighbor assignment at first frame). A number of quantitative metrics are used to gauge performance in terms of quality of detection and tracking, viability of the proposed approach and computation time. Most of these are based on tracking performance metrics used in (Perera et al. 2006). These include,

- *object detection rate* (*ODR*), which is the number of correct detections normalized by number of ground truthed objects. In our experiments, a correct detection is defined as one where the detection has at least a 30 % overlap with the ground truth bounding box,
- *average detection overlap* (*ADO*) represents the mean overlap for only the correct detections, and is at least 0.3,
- *point to point error* (*PPE*) is the mean Euclidean distance between corresponding points in actual and ground truth tracks, regardless of label switching,
- *track fragmentation* (*TF*) counts number of points on a track that actually belong to another track, without normalizing with respect to track length, and
- *track completeness factor* (*TCF*), which measures the percentage of detections that were correctly associated with the corresponding track, and can at most be equal to the object detection rate.

Figure 18a reports the average number of detections per frame for each of the four sequences used in the experiments. The very high number of detections essentially reiterates our previously emphasized point about computational infeasibility of existing approaches, and especially the optimization algorithms. Figure 18b illustrates the average object detection rates for each sequence and verifies the feasibility of the proposed approach used in our framework. Detection rates of more than 90 % were obtained for all sequences. It should be noticed however, that these performances measures are computed by considering the manually ground-truthed objects only. In other words, the false positive detections are not considered in the measure. The reason for this omission is obvious, i.e., counting of false positives requires ground-truthing of each and every object in each frame manually, which is prohibitively cumbersome. It should also be mentioned that although there are some mis-detections using the proposed approach, they do not negatively affect the tracking performance significantly. This is due to the inclusion of detection confidence into the association probability. Therefore, even objects that were mis-detected, but had a non-trivial mean gray area, g_k, (frame difference), can achieve an association likelihood close to or higher than observed detections.

Not withstanding difficulties with detection, probably the most prevalent problem in tracking of large number of objects in dense scenarios is the label switching, i.e., association of detections to wrong tracks. A reasonable metric for performance evaluation measuring label switches is Track Fragmentation. We computed track fragmentation for each ground-truthed track, for each video sequence, and summarized the results in Fig. 19. The figure also illustrates comparison with the baseline Hungarian algorithm tracker, as

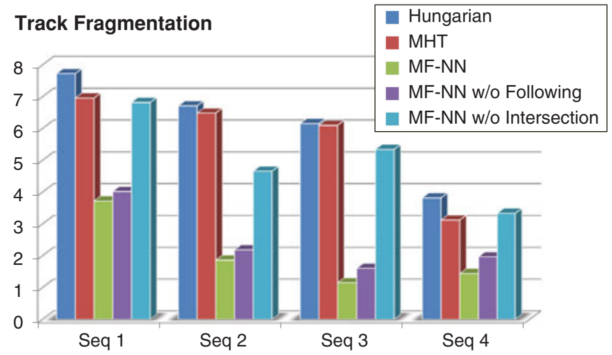


Fig. 19 Influence of context-aware tracking terms, quantified using average track fragmentation for each sequence using: proposed method; bipartite graph matching; and proposed method without vehicle following model, and intersection avoidance. Although the vehicle following term does not drastically improve performance over Hungarian correspondence, the removal of intersection avoidance term significantly increases track fragmentation. See Table 1 for actual values of fragmentation

well as the influence of each of the context-aware tracking cues proposed in this work. First, it can be noticed that the proposed multiframe nearest neighbor tracker significantly outperforms 1–1 instantaneous correspondence algorithm across all sequences. Moreover, it is shown quantitatively that both the context-aware constraints improve performance since when these are removed from the final posterior probability, the number of label switches increases. The vehicle following term, P_f, however, does not seem to make as large an improvement as the track intersection and merging avoidance term, P_m, does. This is intuitive since the intersection and merging avoidance cue performs several functions, not the least of which is to mitigate negative effects of the proposed detection sharing idea, as detailed in Sect. 5.2. As reported in Figs. 18c and 19, the proposed algorithm significantly outperforms bipartite graph matching. Moreover it is also faster by a factor of ∼7. More tracking results and videos are included in supplemental material, showing objects being tracked through occlusions, such as bridges, etc. Occlusions typically persist for about 5–10 frames. Also, many tracked objects do not follow straight or curved constant velocity paths, e.g., vehicles entering or exiting the highway through ramps. The videos show multiple examples of the scenario of abrupt starting and stopping of vehicles when encountering congestion (wave nature of traffic), which is adequately handled by our method. The majority of objects exhibiting

Table 1 Data set specifications and quantitative analysis

Sequence #	1	2	3	4
Camera #	5	4	1, 3	0
Frames	317–416	317–400	10–90	815–915
# GT objects	34	107	50	50
Av. detections/frame	2033	905	1506	1815
Object detection rate	0.9112	0.9837	0.9811	0.9561
Av. detection overlap	0.7913	0.8801	0.7213	0.9571
Point to point Euclidean error (PPE)				
Munkres (Munkres 1957)	26.3057	19.1389	16.1197	21.0928
MHT (Cox and Hingorani 1996)	23.8147	19.9058	11.1270	16.9134
Proposed approach	11.9197	8.1462	7.7572	5.2679
Track completion factor (TCF)				
Munkres (Munkres 1957)	0.4495	0.6943	0.5885	0.4784
MHT (Cox and Hingorani 1996)	0.4763	0.7109	0.6627	0.4954
Proposed approach	0.6430	0.8704	0.7266	0.7889
Track fragmentation (TF)				
Munkres (Munkres 1957)	7.7252	6.7095	6.1538	3.8173
MHT (Cox and Hingorani 1996)	6.9572	6.4854	6.1003	3.1218
Proposed approach	3.7202	1.8648	1.1533	1.4387
Proposed without vehicle following (P_f)	4.0197	2.1837	1.6028	1.9647
Proposed without intersection avoidance (P_m)	6.8194	4.6586	5.3417	3.3327
Computation time/memory				
Time per frame (s)	6.3873	7.6555	8.6281	8.1051
Time per track (ms)	14.9830	9.6873	9.5668	12.2012
Time per candidate (ms)	1.1939	0.6318	0.8735	1.5431
Memory per frame (MB)	19.9623	19.2920	21.6109	21.1410

motion in the scene are successfully detected and tracked throughout the sequences.

We also compared our approach to the well known MHT (Cox and Hingorani 1996) and quantified the results which are reported in Table 1. This evaluation used the same sequences, experimental settings, ground truth, motion model, and detection method, as the proposed framework. At each frame, only the top 10 global hypotheses were retained. As can be seen from the results, the proposed method outperforms MHT in all sequences using all tracking metrics. It does however perform better in one sequence (less Track Fragmentation), if the proposed approach doesn't employ the track intersection and merging avoidance cue (P_m). In general, the performance of MHT is comparable to that of simple bipartite graph matching (Munkres 1957), which is most likely due to the fact that both impose a 1–1 correspondence constraint on tracks and measurements. The main difference between these methods and the proposed approach therefore, is the enumeration of global versus local (object-centric) hypotheses, not one versus multiple hypotheses.

We also attempted to explicitly find the rank of the correct global hypothesis using MHT. For this experiment, we used the correct (ground truth) initialization for the 241 ground-truthed objects, and propagated the corresponding tracks for the first five frames using the true locations, so as to establish the best possible motion model parameters. We observed that the 50 best *global* hypotheses at the sixth frame did not include the correct associations for all 241 objects. Moreover, the *actual best* hypotheses (as per the ground truth) for each of the four sequences were ranked 37, 22, 23, and 14 respectively. The "actual best" hypothesis was computed as the one with the maximum number of correct associations.

It should be kept in mind that for each sequence, *all* detected objects (along with false positive detections) were being tracked, not just the ground truth objects. This is exactly the reason we argue against 1–1 correspondence constraint for tracking in such high density traffic, i.e., the influence of spurious wrong associations in a global hypothesis is difficult to contain, and propagates to neighboring objects, and so on.

Moreover, as mentioned earlier, for this particular experiment, the first five points on the trajectories of ground-truthed objects were manually corrected to avoid disadvantage to the MHT owing to incorrect initialization of motion models. Even though we did not verify these results for all the tracked objects, it is reasonable to conclude from the results of 241 objects that in order to retain the optimal hypothesis, a prohibitively large number of hypotheses will have to be propagated to subsequent frames, as compared to only ten candidates per object in our method. Even large storage and computation power may not completely alleviate the limitations of 1–1 correspondence, because it is not enough to enumerate and retain the correct global hypothesis. It also needs to rank high among the list of possible hypotheses.

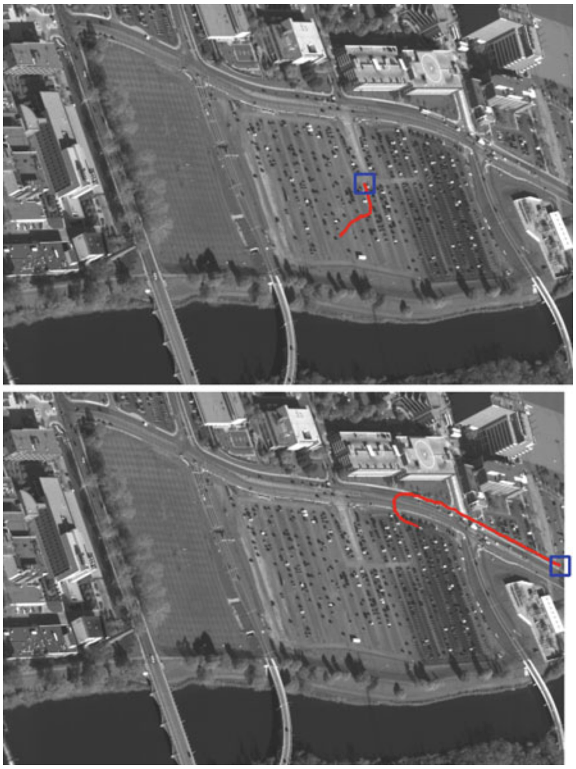


Fig. 20 Example scenarios of vehicles accelerating from rest, making turns, and decelerating and stopping

Tracking in traffic jams and parking lots The constant acceleration motion is greatly helpful in handling move-stop-move scenarios, e.g., in traffic jam scenarios. Tens of examples of almost complete stopping scenarios are shown in the videos uploaded as supplemental material, e.g., in the uploaded video "Sequence1.avi". It should be noted that the videos are being played at 10 fps, which is actually five times the real time speed, since CLIF data is recorded at about 2 fps. Therefore the duration of deceleration and stopping scenarios is longer than it seems.

The scenario where a moving vehicle eventually comes to a halt, e.g., traffic light, is also handled adequately well, again owing to the constant acceleration model, along with hypothetical detection insertion at predicted locations. The maximum number of consecutive hypothetical detections is however restricted, therefore vehicles stopping for longer durations of time are not reacquired. This scenario is actually difficult to observe or test in CLIF imagery since the platform motion precludes persistent observation for long enough periods of time. We do however observe some cases where vehicles initially at rest start moving, e.g., from a parking lot, and enter the road, etc. These scenarios are especially difficult to handle because the vehicles move at very low speeds and

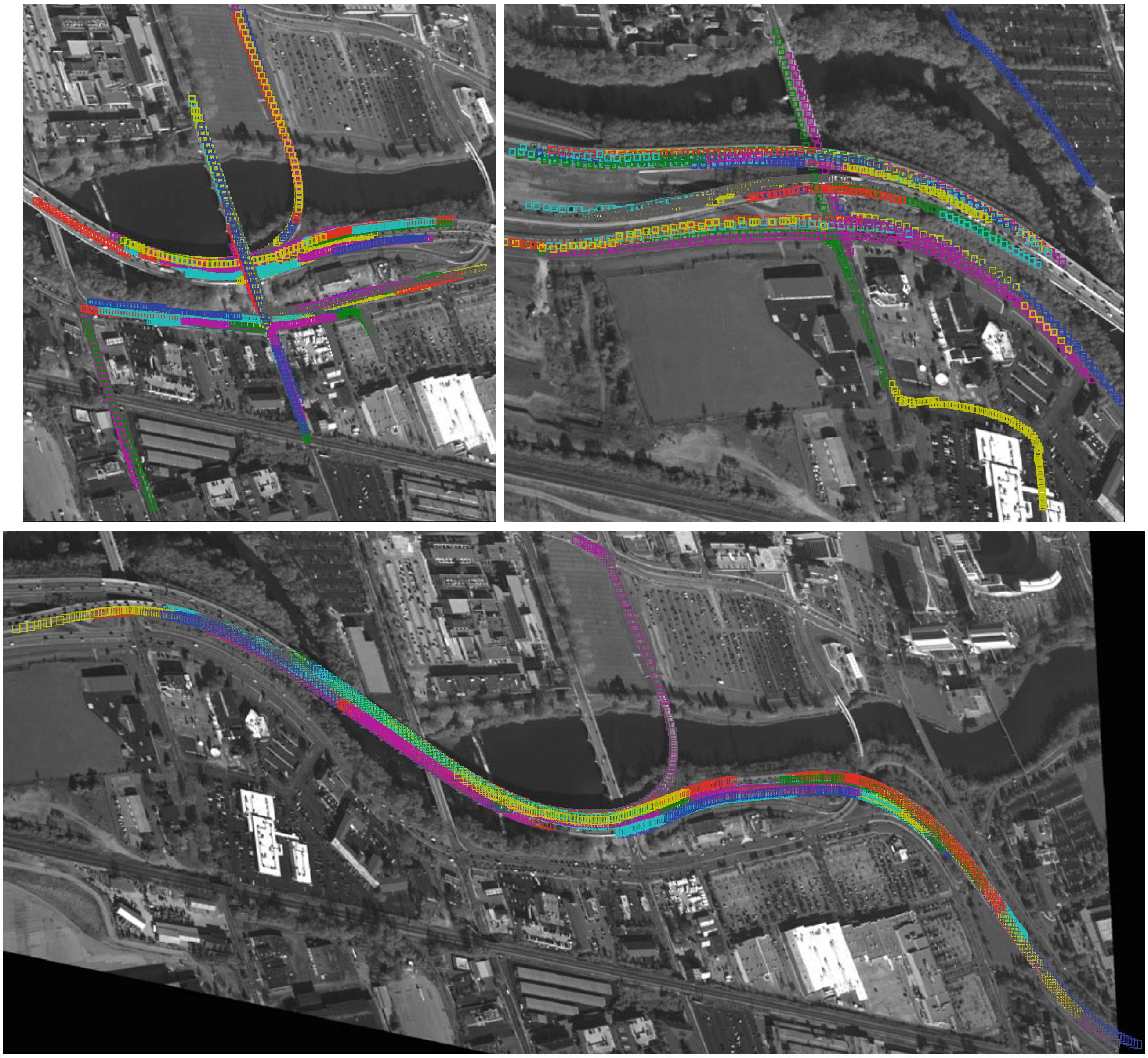


Fig. 21 Tracks obtained by the proposed method shown as sequences of same *color squares*, for three small regions of sequence 2. Notice that tracks are obtained even for vehicles undergoing very long occlusions when driving under the horizontal highway (*left* and *middle* regions). All tracks generated for the sequence are not plotted for clarity (Color figure online)

frequently decelerate or stop before making turns or while waiting to join traffic on the street. Two examples of these cases are shown in Fig. 20.

The case of non-moving vehicles, e.g., parked vehicles, cannot be handled by our method, since the object detection output as well as detection confidence is based on motion (frame difference). Very accurate and computationally efficient vehicle detection algorithms applicable to single images will be required in this case.

Track merging due to detection sharing We also quantified the frequency of occurrence of track merging. For the ground-truthed tracks, the percentage of occurrence of merging, averaged over all 241 objects was 0.83 % (2 out of 241 objects). In the first case, the second candidate for the second track was chosen, and was mostly correct. In the second case, the third candidate for the second track was chosen, because both of the first two candidates merged with the optimal candidate of the first track. This selection process is completely automated, and free of any parameters or thresholds. For all objects detected and tracked in the four selected sequences, the percentage was slightly higher at 1.79 %. These numbers give a coarse estimate of how infrequent eventual track merging is.

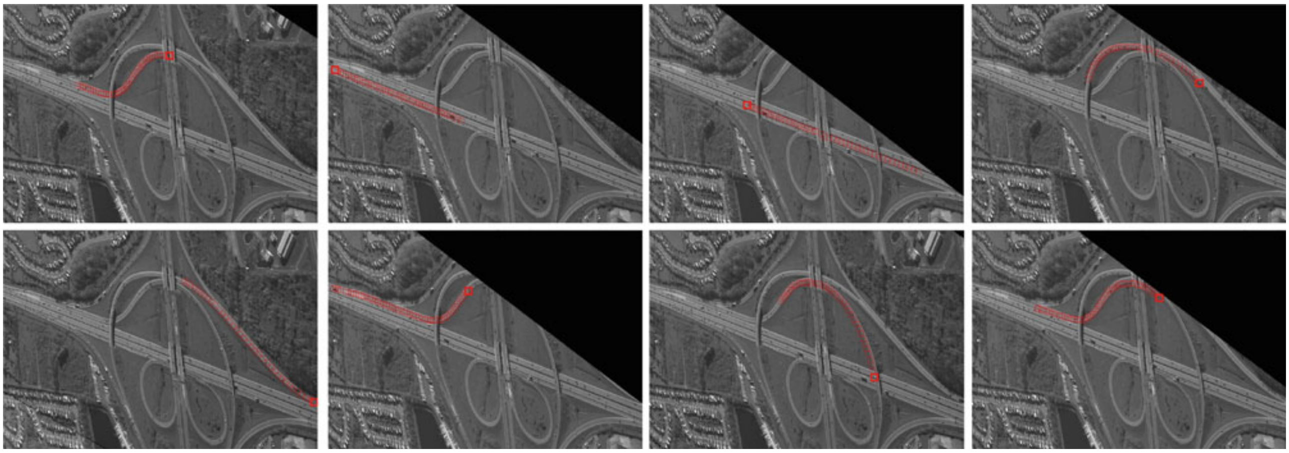


Fig. 22 A few example tracks obtained in Sequence # 3. Each *image* shows track of only one object as a sequence of *red rectangles*. See supplemental material for video (Color figure online)

Finally, to give a bird's eye view of the spatial extent of the scene under consideration, the density of traffic, the variability in object motion, and the duration of object persistence, some of the tracks obtained for two sequences are shown in Figs. 21 and 22.

8 Conclusion

One drawback of the approach is the absence of an explicit 1–1 matching constraint. While we argue that many–many matching provides distinct advantages, the permission of detection sharing sometimes has unexpected consequences. For example, measurement waste is not explicitly forbidden. In other words, some particular candidates of two distinct tracks can start tracking the same object early on during the process, which makes their probabilities from all cues essentially very similar. Even though the track intersection and merging cue kicks in in such scenarios, and the adverse effects were observed to be minimum and rare, the best candidates in distinct tracks having similar probabilities means they will reduce each other's influence by similar amounts. Sometimes this problem may result in duplicate tracks and detection wastage. Such cases are handled during final selection of best candidate.

We also observe that for object-centric hypotheses adverse effects of detection sharing are significantly mitigated due to *discouragement* of track intersection. One of the directions for future research is to devise an optimal association algorithm that explicitly prohibits intersection.

In conclusion, we have presented novel methods of motion detection and MTT of large number of objects in low frame rate, high altitude aerial sequences. Our approach is general and can be applied to any scenario requiring object detection and MTT, which has been validated from experiments on the most challenging aerial surveillance data set available. Our work provides an alternative to global cost minimization frameworks, and is also novel in terms of difficulty of the data set used which is one of the very few publicly available wide area aerial surveillance sequences.

References

Ablavsky, V., Thangali, A., & Sclaroff, S. (2008). Layered graphical models for tracking partially-occluded objects. In *CVPR*, Anchorage.

Bazzani, L., Cristani, M., & Murino, V. (2010). Collaborative particle filters for group tracking. In *ICIP*, Hong Kong.

Berclaz, J., Fleuret, F., Turetken, E., & Fua, P. (2011). Multiple object tracking using k-shortest paths optimization. *IEEE Transactions on Pattern Analysis and Machine*, *33*(9), 1806–1819.

Breitenstein, M., Reichlin, F., Leibe, B., Koller-Meier, E., & Gool, L. V. (2009). Robust tracking-by-detection using a detector confidence particle filter. In *ICCV*, Kyoto.

Cox, I., & Hingorani, S. (1996). An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE TPAMI*, *18*(2), 138–150.

Cox, I., Miller, M., Danchick, R., & Newnam, G. (1997). A comparison of two algorithms for determining ranked assignments with application to multitarget tracking and motion correspondence. *IEEE Transaction of AES*, *33*(1), 295–301.

Cucchiara, R., Piccardi, M., & Mello, P. (2000). Image analysis and rule-based reasoning for a traffic monitoring system. *IEEE Transaction of ITS*, *1*(2), 119–130.

Danchick, R., & Newnam, G. (2006). Reformulating Reid's MHT method with generalised Murty k-best ranked linear assignment algorithm. *IEE Proceedings Radar, Sonar and Navigation*, *153*(1), 22.

Durrant-Whyte, H. (1988). Sensor models and multisensor integration. *International Journal of Robotics Research*, *7*(6), 97–113.

Edie, L. (1960). Car following and steady state theory for non-congested traffic. *Operations Research*, *9*, 66–76.

Gazis, D., Herman, R., & Potts, R. (1959). Car following theory of steady state traffic flow. *Operations Research*, *7*, 499–505.

Grabner, H., Leistner, C., & Bischof, H. (2008). Semi-supervised on-line boosting for robust tracking. In *ECCV*, Marseille.

Hinton, G. (1999). Products of experts. In *Proceedings of the Ninth International Conference on Artificial Neural Networks (ICANN 99)* (pp. 1–6).

Huang, K., Wang, L., Tan, T., & Maybank, S. (2008). A real-time object detecting and tracking system for outdoor night surveillance. *Pattern Recognition*, *41*, 432–444.

Kang, J., Cohen, I., & Medioni, G. (2003). Soccer player tracking across uncalibrated camera streams. In *ICCV: PETS Workshop*, Nice.

Kikuchi, S., & Chakroborty, P. (1992). Car following model based on fuzzy inference system. *Transportation Research Record*, *1365*, 82–91.

Kuhn, H. (1955). The Hungarian method for solving the assignment problem. *Naval Research Logistics Quarterly*, *2*, 83–97.

Mann, S., & Picard, R. (1997). Video orbits of projective group: An approach to featureless estimation of parameters. *IEEE TIP*, *6*, 1281–1295.

Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, *5*(1), 32–38.

Murty, K. (1968). An algorithm for ranking all the assignments in order of increasing cost. *Operations Research*, *16*(3), 682–687.

Nagel, K., & Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. *Journal of Physics I: France*, *2*(12), 2221–2229.

Newell, G. (1961). Nonlinear effects in the dynamics of car following. *Operations Research*, *9*(2), 209–229.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*(1), 62–66.

Perera, A., Srinivas, C., Hoogs, A., Brooksby, G., & Hu, W. (2006). Multi-object tracking through simultaneous long occlusions and split–merge conditions. In *CVPR*, New York.

Pirsiavash, H., Ramanan, D., & Fowlkes, C. (2011). Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR*, Colorado Springs (pp. 1201–1208).

Porikli, F., & Pan., P. (2009). Regressed importance sampling on manifolds for efficient object tracking. *AVSS*, Genoa.

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

Reilly, V., Idrees, H., & Shah, M. (2010). Detection and tracking of large number of targets in wide area surveillance. In *ECCV*, Heraklion.

Schubert, R., Richter, E., & Wanielik, G. (2008). Comparison and evaluation of advanced motion models for vehicle tracking. In *ICIF*, Shanghai.

Schulz, D., Burgard, W., Fox, D., & Cremers, A. (2001). Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *ICRA*, Seoul.

Shafique, K., & Shah, M. (2005). A non-iterative greedy algorithm for multi-frame point correspondence. *IEEE TPAMI*, *27*(1), 51–65.

Shalom, Y., & Fortmann, T. (1988). *Tracking and data association*. London: Academic Press.

Shamos, M., & Hoey, D. (1976). Geometric intersection problems. In *SFCS*, Houston (pp. 208–215).

Song, B., Jeng, T., Staudt, E., & Roy-chowdhury, A. (2010). A stochastic graph evolution framework for robust multi-target tracking. In *ECCV*, Crete.

Stauffer, C., & Grimson, W. (2000). Learning patterns of activity using real-time tracking. *IEEE TPAMI*, *22*(8), 747–757.

USAF. (2006). Columbus large image Format dataset. https://www.sdms.afrl.af.mil/datasets/clif2006/.

Vezzani, R., Baltieri, D., & Cucchiara, R. (2009). Pathnodes integration of standalone particle filters for people tracking on distributed surveillance systems. In *ICIAP*, Vietri sul Mare.

Wang, G., Xiao, D., & Gu, J. (2008). Review on vehicle detection based on video for traffic surveillance. In *ICAL*, Qindao (pp. 2961–2966).

Xiao, J., Cheng, H., Han, F., & Sawhney, H. (2008). Geo-spatial aerial video processing for scene understanding and object tracking. In *CVPR*, Anchorage.

Xiao, J., Cheng, H., Sawhney, H., & Han, F. (2010). Vehicle detection and tracking in wide field-of-view aerial video. In *CVPR*, San Francisco.

Xing, J., Ai, H., & Lao, S. (2009). Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses. In *CVPR*, Miami.

Yang, M., Lv, F., Xu, W., & Gong, Y. (2009). Detection driven adaptive multi-cue integration for multiple human tracking. In *ICCV*, Kyoto.

Yilmaz, A., Javed, O., & Shah, M. (2006). Object tracking: A survey. *ACM Computing Surveys*, *38*, 1–45.

Yin, Z., & Collins, R. (2006). Moving object localization in thermal imagery by forward–backward MHI. In *OTCBVS*, Kokomo.