ROADESIC DISTANCE: FLOW-AWARE TRACKLET ASSOCIATION COST FOR WIDE AREA SURVEILLANCE

Yeti Z. Gürbüz   *Oğul Can*   *A. Aydın Alatan*

Dept. of Elect. and Elec. Eng. & Center for Image Analysis (OGAM)
Middle East Technical University, Ankara, Turkey

ABSTRACT

Long-term multi-target tracking via tracklet merging in wide area surveillance has crucial importance to improve tracker performances and operational requirements. Min-cost network flow formulation for multi-target tracking is adopted for the tracklet merging problem. In order to improve the continuity of the computed flows by the min-cost network flow framework, a novel tracklet association cost is proposed to be utilized in this network. The proposed cost is based on connecting two tracklets by considering the traffic flow which is estimated from the precomputed tracklets. Such an approach enforces spatial consistencies between tracks by imposing these relations into the association cost. Hence, without violating the min-cost network flow formulation, a constraint to enforce spatial consistency can be implicitly obtained. The proposed cost function can be further exploited to interpolate gaps between the merged tracklets for post-processing. The experimental results show that proposed association cost improves baseline framework that uses costs considering only two tracklets at a time, as well as some other tracklet merge algorithms from the literature.

Index Terms— Multi-target tracking, tracklet merging, wide area surveillance, roadesic distance

1. INTRODUCTION

Multi-target tracking (MTT) is one of the crucial tasks in wide area surveillance (WAS) and has additional challenges compared to traditional target tracking problems in computer vision [1]. WAS imagery in general lacks of color information and the target size is typically much smaller due to the altitude of the capturing sensor. Those constraints together make appearance and shape-based tracking methods inapplicable to the problem. Moreover, owing to the size of the observed area, the computational burden of the algorithms is quite high to be solved by sophisticated tracking methods.

The nature of the MTT in WAS problem moves the general tendency towards developing hierarchical methods [2–9] in which moving object detection at each frame is followed by association of them across frames to form tracklets (i.e. short tracks) to be merged to have long-term tracks. Bipartite graph matching with Hungarian algorithm [10, 11] is exploited in [2, 3, 5, 8] to associate detections and tracklets. Association is improved by introducing split and merge conditions of the tracklets in [3]. Tile processing is proposed in [5] to reduce the problem size and utilize graph matching in detection association level as well. Spatial constraints are considered during tracklet formation in [8] to have tracklets that obey traffic flow. Reliable tracklet extraction is addressed in [7] by formulating the problem as inference in Bayesian networks and improved in [9] by introducing spatial constraints during inference. Similarly, merging tracklets through a regression tracker to improve continuity of the tracks is proposed in [6]. Instead of global association of the detections or traclets via graph matching, local comparison and matching of the tracklets [4] or detections [3, 12] is also considered. Dynamic programming is used to extract tracklets by utilizing the motion of the detections in [4] and the extracted tracklets are merged according to consideration of pairwise similarities. Kalman filtering is performed in [12] for multiple hypothesis tracking (MHT). Similar to filtering, nearest neighbor based association is utilized in [13] with many to many correspondence relaxation to the MHT problem.

The aforementioned approaches are greedy in terms of time expanse. Namely, very short span of time is considered during association and effects of long-term relations among tracks are ignored. On the other hand, the approaches that follow min-cost flow on a directed graph framework for MTT [14–18] are promising for long-term association of the both detections [15, 16] and tracklets [17], since the association problem is solved globally in space and time. The reason for not selection of those methods in WAS applications is due to the fact that the edge costs of the graphs play important role on the performance. While improvement of the edge costs by metric learning is propose in [17], proper edge costs cannot be assigned in WAS scenarios due to lack of shape and appearance information.

This study aims to adapt min-cost flow framework to WAS by proposing a novel tracklet association cost that enforces spatial consistency among tracklets. Unlike the methods [14, 15] that change the nature of the min-cost flow

This study is supported by ESEN System Integration.

problem by introducing constraints to impose spatial consistencies, the proposed method implicitly enforces spatial consistencies via novel association cost without modifying the min-cost flow problem.

2. PROPOSED METHOD

The proposed method takes precomputed tracklets as input and yields longer tracks by merging them. The proposed merging framework is based on computing min-cost flows on a directed graph and the edges of this graph are weighted according to the proposed novel cost function which enforces spatial consistencies among tracks, as well as fills the gaps between the merged tracklets.

2.1. Graph-based Tracklet Merging Framework

Adopted from [16], the tracklet merging problem is to be formulated as a min-cost flow problem on a directed graph similar to [17]. The graph is constructed by tracklets as the vertices, V and the possible connections among the tracklets as the edges, E. A tracklet i, in its most simple form, is a collection of two tuples: $\tau_i = \{(\mathbf{p}_n^i, t_n^i)\}_{n=1}^{N_i}$ where $\mathbf{p} = [x\, y]^T$ is position and t is the time instant-frame number- of the position. Therefore, a tracklet i of duration N_i frames has a begin- and an end-time, denoted as $t_{bgn}^i = t_1^i$ and $t_{end}^i = t_{N_i}^i$ respectively, and it has a begin and an end position denoted as $\mathbf{p}_{bgn}^i = \mathbf{p}_1^i$ and $\mathbf{p}_{end}^i = \mathbf{p}_{N_i}^i$, respectively. A binary variable $e_{ij} = \mathbb{1}_{\mathbb{Z}^+}(t_{bgn}^j - t_{end}^i)$ denotes a directed edge, (i, j), from the tracklet vertex i to the tracklet vertex j with indicator function $\mathbb{1}_{\mathbb{Z}^+}(\cdot)$ so that a tracklet cannot be connected to a tracklet as long as their existences overlap in time. The graph is augmented by two vertices, $\mathcal{V} = V \cup \{v_s, v_p\}$ standing for starting and stopping, respectively. These two vertices are connected to all the vertices representing tracklets: $e_{v_s j} = 1, e_{j v_p} = 1\, \forall j \in V$. Hence, final edge set becomes $\mathcal{E} = E \cup \{(v_s, j)\} \cup \{(j, v_p)\}\, \forall j \in V$, namely, $\mathcal{E} = \{(i, j) \mid i, j \in \mathcal{V} \text{ and } e_{ij} = 1\}$.

Once the graph, $(\mathcal{V}, \mathcal{E})$, is constructed, each edge, $(i, j) \in \mathcal{E}$ is associated with a cost, c_{ij}, and a binary variable, s_{ij}, that indicates whether the corresponding edge from vertex i to j is selected to indicate merger between two tracklets. Let $\mathbf{s}$ be the decision vector entities of which are s_{ij} variables. Hence, tracklet merging problem can be formulated as:

$$\mathbf{s}^* = \underset{\mathbf{s}=[\cdots s_{ij} \cdots]^T}{\text{argmin}} \sum_{(i,j)\in\mathcal{E}} s_{ij}\, c_{ij} \tag{1a}$$

$$\text{s. t. } s_{ij} \in \{0, 1\} \qquad \forall (i, j) \in \mathcal{E} \tag{1b}$$

$$\sum_{j\in\mathcal{N}(i)} s_{ij} = 1 \qquad \forall i \in \mathcal{V} \setminus \{v_s, v_p\} \tag{1c}$$

$$\sum_{j\in\mathcal{N}(i)} s_{ij} - \sum_{k: i\in\mathcal{N}(k)} s_{ki} = 0 \qquad \forall i \in \mathcal{V} \setminus \{v_s, v_p\} \tag{1d}$$

$$\sum_{j\in\mathcal{N} v_s)} s_{v_s j} - \sum_{k: v_p\in\mathcal{N}(k)} s_{k v_p} = 0 \tag{1e}$$

where $\mathcal{N}(i) = \{j \mid e_{ij} = 1\}$ is the neighborhood of vertex i. Constraint (1c) assures that a tracklet can be merged with only one tracklet and constraints (1d) and (1e) are to provide continuous connections from start vertex to stop vertex. The problem in (1) is a min-cost network flow problem [18] adopted to tracklet merging in WAS; hence, the resultant flows of $\mathbf{s}^*$ from start to stop vertex are the estimated tracks from the tracklets.

In this off-line formulation, all the available tracklets are merged through a global consideration and the optimality of the computed tracks are based on the edge costs, c_{ij}. Proper edge costs are required to match the optimal tracks with the desired tracks. The next subsection is devoted to that problem through proposing a novel association cost.

2.2. Proposed Tracklet Association Cost

The main goal is to define a cost function that imposes consistency among the spatial neighboring tracks in addition to assigning an association cost between two tracklets according to their similarity. The consistency among the spatial neighboring tracks can be directly forced by adding a related constraint to the problem defined in (1). Yet, the problem will no longer be a min-cost network flow problem and cannot be solved by efficient methods [15]. Thus, the aim of proposing an association cost is to enforce spatial consistency implicitly by fusing it into the edge cost, c_{ij}, so that the nature of the optimization problem in (1) is not violated.

To formulate the proposed cost, firstly, the vehicle flow information, which is direction and speed at the spatial locations, is to be extracted from the precomputed T tracklets, $\{\tau_i\}_{i=1}^T$. Only direction of the flow is considered, since the vehicle speed is assumed to be uniformly distributed over the whole range at each location in general. Based on the estimated velocities, $\mathbf{v}_n$, from the positions of consecutive detected positions, $(\mathbf{p}_n, \mathbf{p}_{n+1})$, each spatial location in the scene is assigned to some directions after clustering the direction information from the tracklets. A tracklet provides direction information for the pixels of the detections and for the pixels along the line between two consecutive detection positions by assigning the direction in between. K-means clustering is performed at each pixel location by iteratively increasing K, when a direction that is at least $\pi/16$ apart from the existing cluster centers is observed. Thus, a direction set, $\mu_\phi(\mathbf{p})$ of $K_\mathbf{p}$ directions is obtained for each location:

$$\mu_\phi(\mathbf{p}) = \{\phi_1, \ldots, \phi_{K_\mathbf{p}}\} \tag{2}$$

Once the flow information is obtained (Fig. 1a), the next step is modeling the vehicle motions. Vehicles are assumed to be controlled by constant steering, a^ϕ, and acceleration, $a^\mathcal{S}$, within Δt time duration. Thus, a vehicle state $\mathbf{x}_n = [\mathbf{p}_n\, \phi_n\, \mathcal{S}_n]^T$ with position, direction and speed has the state transition of the form:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{g}(a_n^\phi, a_n^\mathcal{S}, \phi_n, \mathcal{S}_n, \Delta t) \tag{3}$$

where $\mathbf{g}(\cdot)$ is such that:

$$\begin{aligned} [\phi_{n+1}\, \mathcal{S}_{n+1}]^T &= [\phi_n\, \mathcal{S}_n]^T + \Delta t\, [a_n^{\phi}\, a_n^{\mathcal{S}}]^T \\ \mathbf{p}_{n+1} &= \mathbf{p}_n + \tfrac{\Delta t}{2}\left(\mathcal{S}_n\, \mathbf{u}(\phi_n) + \mathcal{S}_{n+1}\, \mathbf{u}(\phi_{n+1})\right) \end{aligned} \tag{4}$$

with $\mathbf{u}(\phi) = [cos(\phi)\, sin(\phi)]^T$. Given two tracklets, τ_i and τ_j, to be associated, $N{=}\frac{t_{bgn}^j - t_{end}^i}{\Delta t}$ controls are required to reach $\mathbf{x}_{bgn}^j$ from $\mathbf{x}_{end}^i$. The tracklets i and j are considered to be associated, if there exists a set of controls, $\{a_n^{\phi}, a_n^{\mathcal{S}}\}_{n=0}^{N-1}$, such that $\mathbf{x}_{bgn}^j$ is reachable from $\mathbf{x}_{end}^i$ as the intermediate states, $\mathbf{x}_n$, obey the global flow behavior. Formally, the association cost, C_{ij}, can be formulated as:

$$C_{ij} = \min_{\{a_n^{\phi}, a_n^{\mathcal{S}}\}_{n=0}^{N-1}} \|\mathbf{p}_{bgn}^j - \mathbf{p}_N\|^2 \tag{5a}$$

$$\text{s. t.} \quad \phi_N = \phi_{bgn}^j \tag{5b}$$

$$\min_{\phi \in \mu_\phi(\mathbf{p}_n)} |\phi - \phi_n| = 0 \text{ for } n = 1, \ldots, N{-}1 \tag{5c}$$

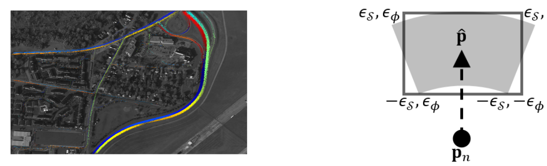
$$|a_n^{\phi}| \leqslant \epsilon_\phi,\ |a_n^{\mathcal{S}}| \leqslant \epsilon_{\mathcal{S}} \text{ for } n = 0, \ldots, N{-}1 \tag{5d}$$

where $\mu_\phi(\cdot)$ is given by (2) and state variables, $\mathbf{p_n}, \phi_n, \mathcal{S}_n$, are obtained through (3) with $\mathbf{x}_0{=}\mathbf{x}_{end}^i$. $\mathbf{p}_N$ is the final position after N^{th} control and is a function of $\{a_n^{\phi}, a_n^{\mathcal{S}}\}_{n=0}^{N-1}$. Constraint (5b) is to match the final directions and constraint (5c) forces the intermediate states to have a direction according to vehicle flow. Constraint (5d) restricts the magnitude of the controls with ϵ_ϕ and $\epsilon_{\mathcal{S}}$ to prevent improbable actions.

The problem given by (5) is to be solved approximately via a heuristic search under mild assumptions. The problem can be simplified by assuming a constant average speed, $\mu_{\mathcal{S}}^{ij} = (\|\mathbf{v}_{end}^i\| + \|\mathbf{v}_{bgn}^j\|)/2$, in all states, since the aim is not to find the true actions but to determine whether a spatially consistent path exists between two tracklets. Now, the problem is finding a set of steerings to reach $\mathbf{x}_{bgn}^j$ as considering spatial consistency. For this purpose, a heuristic search is always performed with $a_n^{\mathcal{S}}{=}0$.

2.3. Computation of the Proposed Cost

Starting from the initial state, $\mathbf{x}_0{=}\mathbf{x}_{end}^i$, the next position, $\hat{\mathbf{p}}$, is obtained by (4) with no control, $a_0^{\phi}{=}0$, at fixed speed, $\mu_{\mathcal{S}}$. A rectangular region which approximately realizes the constraints in (5d) is defined around $\hat{\mathbf{p}}$, as in Fig. 1b. Within the region, for each pixel location, $\hat{\mathbf{p}}_k$, which has direction information, the required steering control, $a_{\hat{\mathbf{p}}_k}^{\phi}{=}\frac{1}{\Delta t}(\mu_\phi(\hat{\mathbf{p}}_k){-}\phi_0)$, is determined for each direction in the set, $\mu_\phi(\cdot)$. If the steering is within allowed range, $|a_{\hat{\mathbf{p}}_k}^{\phi}| \leqslant \epsilon_\phi$, then the position error, $\varepsilon = \|\hat{\mathbf{p}}_k - \hat{\mathbf{p}}_{0\to 1}\|$, is examined, where $\hat{\mathbf{p}}_{0\to 1}$ is the next position from $\mathbf{p}_0$ with steering $a_{\hat{\mathbf{p}}_k}^{\phi}$ at speed $\mu_{\mathcal{S}}$ according to (4). Location $\hat{\mathbf{p}}_k$ and its tested direction are added to a candidate state queue, if the positional error is insignificant, $\varepsilon \leqslant \delta$. Here, δ is to allow small perturbations to average speed to match with the directions as in (5c). After all locations, $\hat{\mathbf{p}}_k$, are processed, the candidate state that has the



(a) Flow info. $\mu_\phi(\cdot)$ in (2) (b) Approximation of (5d)

Fig. 1: (a) Coloring is according to angle. (b) Rectangular window is used to approximately satisfy constraints in (5d).

direction closest to the target, ϕ_{bgn}^j, is selected for the following steps. This aforementioned process is repeated in a depth-first-search manner for N steps until the target position, $\mathbf{p}_{bgn}^j$, lies within the rectangular region of the state, $\mathbf{x}_{N-1}$, or until all the candidates are processed. Once the search is to be terminated, the positional error, ε_f, with the target is taken as the cost:

$$C_{ij} = \varepsilon_f = \|\mathbf{p}_{bgn}^j - \hat{\mathbf{p}}_{N-1\to N}\|,\ \hat{\mathbf{p}}_{N-1\to N} = \mathbf{f}(\mathbf{p}_{bgn}^i) \tag{6}$$

where the steering for the state transition should satisfy (5b).

In practice, if the scene is observed enough, almost every location will have a flow information. However, if there is no direction information at any stage of the search, the difference between the current and the target direction is naively divided into the remaining number of steps and the resultant steering is applied as the control.

The proposed method assigns low costs to the association of the tracklets that are reachable via following the flow of the underlying road, while highly penalizing the ones that are not reachable. The approximate solution of the problem in (5) considers the global tendency of the location at each step and thus the exploiting the obtained cost as the edge cost results in imposing the spatial consistency implicitly for the problem in (1).

When the trail of the intermediate states of the solution of (5) is considered, the path in between two tracklets resembles the geodesic path through the roads with direction; therefore, the proposed distance is named *roadesic distance* . It should be noted that the proposed roadesic distance calculation not only provides a cost metric but also an interpolation method to complete the gaps between tracklets after merging.

2.4. Edge Costs

For the edges between tracklets, the cost is proportional to the proposed roadesic distance, C_{ij}, in (6):

$$c_{ij} = \frac{C_{ij}}{\sigma_d},\ \forall (i,j) \in \mathcal{E},\ i \neq v_s,\ j \neq v_p \tag{7}$$

where σ_d is for normalization. For the edges connected to start v_s and stop v_p vertices, the cost should be proportional to the speed of the vehicles, $\mathcal{S}^k$, with appropriate normalization, $\sigma_{\mathcal{S}}$, to avoid from unlikely appearance/disappearance events

Table 1: Evaluation of the performance metrics with the baseline and proposed method.

Method:	**Unmerged Raw Tracklets/Baseline/Proposed**							
	D-Pd	**D-Pfa**	**Tr-Pd**	**Tr-Pfa**	**Tr-Cont**	**Tr-Pur**	**Tar-Cont**	**Tar-Pur**
EsTrl [2]	.54/.54/**.58**	.87/.86/.86	.94/.94/.94	.18/.20/.21	1.07/1.30/1.25	.98/.91/**.93**	2.02/1.44/**1.39**	.52/.57/**.61**
ExTrl1	.56/.56/**.68**	.01/.01/.05	.99/.99/.99	.04/.02/.02	1.05/1.22/1.30	1.00/**.99**/.98	3.94/**1.26**/1.29	.27/.67/**.88**
ExTrl2	.53/.53/**.66**	.01/.00/.04	.99/.99/.99	.04/.02/.01	1.04/1.23/1.28	1.00/.95/**.97**	3.77/2.14/**1.68**	.38/.46/**.70**

and trivial solutions composing of single tracklet.

$$c_{v_s k} = \frac{\mathcal{S}^k_{bgn}}{\sigma_{\mathcal{S}}},\ c_{kv_p} = \frac{\mathcal{S}^k_{end}}{\sigma_{\mathcal{S}}},\ \forall k \in \mathcal{V} \setminus \{v_s, v_p\} \qquad (8)$$

However, since entrance and exit to the scene might happen, the tracklets moving towards to or from the image boundary have fixed moderate connection costs, i.e. $C_{i/o}$ for in and out.

3. EXPERIMENTS

Proposed method is tested on the WPFAB 2009 dataset [19] to provide quantitative results. The dataset provides two test data which are tracklets extracted from the ground truth tracks (ExTrl1) and the estimated tracklets with the method in [2] (EsTrl). One more test data which has larger gaps between tracklets (ExTrl2) is extracted manually from the ground truth tracks to realize more challenging scenarios.

Throughout the experiments, the parameters are fixed at Δt=.5, δ=16, σ_d=16, $\sigma_{\mathcal{S}}$=24, $C_{i/o}$=$-log(.65)$ with 20×20 roadesic search window that implicitly defines ϵ_ϕ (Fig. 1). The spatial and temporal units are pixels and frames, respectively. The parameters are set according to $.25m$ ground sampling distance and $1.25Hz$ frame rate.

For comparison, a baseline framework that exploits the greedy kinematic cost presented by [2, 4, 8] as the edge costs in (1) is considered. The evaluation metrics for the comparison are explained in [20], where target continuity (Tar-Cont) and track purity (Tr-Pur) are the most significant metrics to evaluate merging performance. The aim is to reduce Tar-Cont to 0, indicating that the truth track is composed of a single track, while keeping Tr-Pur at 1 as much as possible, namely, true associations among tracklets should be performed. Target purity (Tar-Pur) measures the proportion of the truth detections of a truth track that overlaps with a computed track. Tar-Pur close to 1 is desired and merging tracklets improves the performance on this metric to some degree. It should be noted that interpolation is required for significant improvement. Probability of true detection (D-Pd) is the percentage of the detected ground truth detections. The comparative evaluation of the aforementioned metrics with the proposed and the baseline framework are provided in Table 1. In this table, evaluation of raw unmerged tracklet results by using the given metrics are also provided to give a lower bound on the performances.

Table 1 indicates that the proposed method improves Tar-Cont with minimum degradation on Tr-Pur in challenging scenarios where either the tracklets are sparse or estimated by [2]. For the dataset that has tracklets with small gaps, the baseline framework slightly performs better. In all scenarios, the proposed method improves Tar-Pr and D-Pd owing to the roadesic interpolation of the gaps between merged tracklets (Fig. 2). Finally, when the reported results of [2] on [19] is considered, utilizing the proposed framework at merging step improves the method in [2].

Fig. 2: A sample track (solid+dashed brown) by merging tracklets (solid brown) with roadesic interpolation (dashed brown) and its ground truth track (solid green).

4. CONCLUSION

In this study, a multi-target tracking method via merging the precomputed tracklets in WAS is presented. The proposed algorithm is adopted from the graph based approach in which min-cost network flow problem is formulated to have merged tracklets as the flows. A novel tracklet association cost is proposed to improve the performance and quality of the computed tracks. The proposed cost, namely roadesic distance, makes use of the global information contributed by the all observed tracklets. Thus, it implicitly imposes a spatial consistency constraint among tracks without explicitly adding a related constraint to the min-cost flow problem. The computation of the roadesic distance inherently interpolates the gaps between merged tracklets. The experimental results validates that the proposed cost improves baseline framework in both merging performance and track completion by interpolation sense. Finally, it should be noted that the proposed method does not utilize target intensities, since gray-valued information is typically not reliable. If the sensor capabilities permit, color or shape can also be integrated easily into the proposed solution.

5. REFERENCES

[1] Alper Yilmaz, Omar Javed, and Mubarak Shah, “Object tracking: A survey,” *Acm computing surveys (CSUR)*, vol. 38, no. 4, pp. 13, 2006.

[2] Arslan Basharat, Matt Turek, Yiliang Xu, Chuck Atkins, David Stoup, Keith Fieldhouse, Paul Tunison, and Anthony Hoogs, “Real-time multi-target tracking at 210 megapixels/second in wide area motion imagery,” in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*. IEEE, 2014, pp. 839–846.

[3] AG Amitha Perera, Chukka Srinivas, Anthony Hoogs, Glen Brooksby, and Wensheng Hu, “Multi-object tracking through simultaneous long occlusions and split-merge conditions,” in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. IEEE, 2006, vol. 1, pp. 666–673.

[4] Bor-Jeng Chen and Gérard Medioni, “Motion propagation detection association for multi-target tracking in wide area aerial surveillance,” in *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*. IEEE, 2015, pp. 1–6.

[5] Vladimir Reilly, Haroon Idrees, and Mubarak Shah, “Detection and tracking of large number of targets in wide area surveillance,” in *European Conference on Computer Vision*. Springer, 2010, pp. 186–199.

[6] Jan Prokaj and Gérard Medioni, “Persistent tracking for wide area aerial surveillance,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1186–1193.

[7] Jan Prokaj, Mark Duchaineau, and Gérard Medioni, “Inferring tracklets for multi-object tracking,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*. IEEE, 2011, pp. 37–44.

[8] Jiangjian Xiao, Hui Cheng, Harpreet Sawhney, and Feng Han, “Vehicle detection and tracking in wide field-of-view aerial video,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 679–684.

[9] Jan Prokaj, Xuemei Zhao, and Gérard Medioni, “Tracking many vehicles in wide area aerial surveillance,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*. IEEE, 2012, pp. 37–43.

[10] Harold W Kuhn, “The hungarian method for the assignment problem,” *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[11] Chang Huang, Bo Wu, and Ramakant Nevatia, “Robust object tracking by hierarchical association of detection responses,” in *European Conference on Computer Vision*. Springer, 2008, pp. 788–801.

[12] Mark Keck, Luis Galup, and Chris Stauffer, “Real-time tracking of low-resolution vehicles for wide-area persistent surveillance,” in *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*. IEEE, 2013, pp. 441–448.

[13] Imran Saleemi and Mubarak Shah, “Multiframe many–many point correspondence for vehicle tracking in high density wide area aerial videos,” *International journal of computer vision*, vol. 104, no. 2, pp. 198–219, 2013.

[14] Hao Jiang, Sidney Fels, and James J Little, “A linear programming approach for multiple object tracking,” in *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[15] Asad A Butt and Robert T Collins, “Multi-target tracking by lagrangian relaxation to min-cost network flow,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1846–1853.

[16] Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua, “Multiple object tracking using k-shortest paths optimization,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.

[17] Bing Wang, Gang Wang, Kap Luk Chan, and Li Wang, “Tracklet association with online target-specific metric learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1234–1241.

[18] Hamed Pirsiavash, Deva Ramanan, and Charless C Fowlkes, “Globally-optimal greedy algorithms for tracking a variable number of objects,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1201–1208.

[19] AFRL, “Air Force Research Labs. SDMS: WPAFB 2009 Dataset.,” `https://www.sdms.afrl.af.mil/index.php?collection=wpafb2009`, 2009.

[20] Roddy Collins, “PPAML WAMI evaluation metrics,” `http://ppaml.galois.com/wiki/raw-attachment/wiki/CP3WAMITrackLinking/WAMIEvaluationMetrics.pdf`, 2014.