FIRST-PERSON MULTIPLE OBJECT TRACKING IN COMPLEX TRAFFIC SCENES

Tingting Jiang, Zhao Zhang, Yuansheng Xu, Yichong Bai, Yizhou Wang

National Engineering Lab for Video Technology,
Key Lab of Machine Perception(MoE),
School of EECS, Peking University, Beijing, China

ABSTRACT

In this paper, we study multi-object tracking problem from the first-person viewpoint,e.g., the moving camera. This problem is different from the traditional one with static camera and brings lots of challenges. To solve this problem, we adopt the tracking-by-detection approach and design a new similarity model for two detection responses considering the camera motion. The similarity model can handle the change of scale and position of objects under the movement of camera. We also consider the detection prior and appearance to improve the tracking performance. The final tracking problem is solved within a network flow framework. Experimental results on KITTI dataset demonstrate the advantages of our method.

Index Terms— multi-object tracking, network flow, visual odometry

1. INTRODUCTION

Tracking multiple objects in complex traffic scenes is of great interest for traffic management and intelligent transportation. Many existing tracking approaches reply on the assumption that cameras are static, which is surveillance. However, tracking multiple objects from the first-person viewpoint, e.g., cameras are installed on the vehicles, is also very important, esp. for developing autonomous vehicles and understanding the drivers' behavior. This problem brings many new challenges compared to standard multi-object tracking problems. Because the cameras can move fast and change direction abruptly, the objects under tracking can appear and disappear from time to time. The illumination and appearance changes can be large. In complex traffic scenes, the background is dynamic and clutter. All these new challenges make previous tracking approaches relying on initialization such as the early work [1] fail on this problem.

In recent years, with the improvement of object detectors [2], tacking-by-detection approaches become the most popular methods for tracking multi-objects [3, 4]. These methods usually apply a pre-trained object model to generate object candidates in each fame, and then associate these candidates across frames to generate the trajectory of each object. They are helpful in the complex traffic scenes we are encountering. To associate object candidates, the similarity model is crucial. [5, 6, 7, 8] measure the similarity between candidates according to their location, scale and appearance. Veenman et al. [5] find optimal bipartite matchings between every two consecutive frames and link them into global trajectories. However, this method only uses the information from two consecutive images so that the target interaction or occlusion would be intractable [9]. To overcome this weakness, Zhang et al. [6] build a network that incorporate all pair-wise similarity information of the whole image sequence. The optimal assignment can be obtained by solving a min-cost flow problem [10] which can be further accelerated by successive shortest path algorithm or two-pass dynamic programming approximation [7, 8]. Some researchers try to introduce motion information to similarity model such as velocity [11, 12, 9]. This requires the similarity function being high order, which makes the optimization NP hard [12]. Collins [11] employs the similarity model which involves data from two or more frames and apply a block ICM (iterated conditional modes) based algorithm to approximate the optimal solution. Butt and Collins [9] build a network where nodes are pairs of candidates in two consecutive frames, so that edges can represent the velocity similarity. They solve it by applying Lagrangian relaxation to the objective and using network flow as subroutine. However the above methods are not suitable for first-person multi-object tracking problem because none of them consider the dynamics of cameras specifically.

In this paper, we tackle the first-person multiple object tracking problem by considering the dynamics of cameras. Based on the network flow framework [6], we design a new similarity model making use of the camera velocity estimated by visual odometry [13]. This new similarity term can reflect the correlation between the object changes and the velocity of cameras. For example, when the camera moves fast, the changes of position or scale of still objects are expected to be larger than the cases when the camera moves slow. Specifically, this correlation can be learned from the training data and then applied on test data. Unlike the previous work [11] and [9] which assume that the objects have constant velocity, we do not make any assumptions about the camera motion.

Besides the new similarity model, we use a new detection prior and a new appearance model in order to improve the tracking performance. The proposed method has been tested on the KITTI [14] dataset and the results show that it can outperform the previous methods.

The rest of the paper is organized as follows. Section 2 introduces our method in details and Section 3 shows the tracking results. And finally Section 4 concludes the paper.

2. METHOD

2.1. Problem Formulation

Given a video sequence, let $\mathcal{X} = \{\mathrm{x}_i\}$ represent the detection results for one specific object class, such as car or pedestrian. Each x_i is a detection response. Here $\mathrm{x}_i = \{x_i, s_i, a_i, t_i\}$ represents center position, height, appearance, frame number of detection response. Let $V = \{d_i\}$ denote the estimated camera motion of the whole video sequence by visual odometry [13], where d_i represents the displacement from frame i to $i+1$. A single trajectory is defined by an ordered list of detection responses, $T_m = \{\mathbf{x}_{k_1}, \mathbf{x}_{k_2}, ..., \mathbf{x}_{k_{l_m}}\}$. Here $\mathbf{x}_{k_i} \in \chi$. l_m is the length of T_m. The set of single trajectories in a video sequence is trajectory hypotheses, i.e. $\mathcal{T} = \{T_m\}$. If we choose the best hypotheses in a Maximum a posteriori estimation (MAP) way, similar to the framework of [6], we have the following formulation of getting an optimal tracking hypothesis $\mathcal{T}$ from observation $\mathcal{X}$ and estimated camera motion V:

$$\begin{aligned}\mathcal{T}^* =& \arg\max_{\mathcal{T}} P(\mathcal{T}|\mathcal{X}, V) \\ =& \arg\max_{\mathcal{T}} P(\mathcal{X}|\mathcal{T}, V)P(\mathcal{T}|V) \\ =& \arg\max_{\mathcal{T}} \prod_i P(\mathrm{x}_i|\mathcal{T}, d_i) \prod_{T_m \in \mathcal{T}} P(\mathcal{T}_m|V) \end{aligned} \quad (1)$$

and

$$\begin{aligned}P(\mathcal{T}_m|V) &= P(\{\mathrm{x}_{k_1}, \mathrm{x}_{k_2}, ..., \mathrm{x}_{k_{l_m}}\}|V) \\ &= P_{enter}(\mathrm{x}_{k_1})P_{link}(\mathrm{x}_{k_2}|\mathrm{x}_{k_1}, V) \\ &\quad ...P_{link}(\mathrm{x}_{k_{l_m}}|\mathrm{x}_{k_{l_m-1}}, V)P_{exit}(\mathrm{x}_{k_{l_m}})\end{aligned} \quad (2)$$

2.2. Transition Probability

The transition probability term $P_{link}(\mathrm{x}_i|\mathrm{x}_j, V)$ can be factorized based on independent assumption as follows:

$$\begin{aligned}P_{link}(\mathrm{x}_j|\mathrm{x}_i, V) = P(x_j|x_i, s_i, V, \Delta t) * P(s_j|x_i, s_i, V, \Delta t) \\ * P(a_j|a_i) * P(\Delta t)\end{aligned} \quad (3)$$

We assume $P(x_j|x_i, s_i, V, \Delta t)$ follows a normal distribution $N(\bar{x}_j, \sigma^2)$. Considering that the object position in the camera coordinate is affected by camera motion and therefore x_j is also affected, we make use of 3D inference to help estimate the mean of the normal distribution. We firstly reconstruct the 3D position x_i^{3D} of detection response x_i in camera coordinate by using the information of detection response, camera calibration and object height [15]. The object height is learned globally from training data. Then we estimate its new 3D position in frame j as

$$x_j^{3D} = x_i^{3D} + \sum_{k=i}^{j-1} d_k. \quad (4)$$

Finally the estimated 2D position of x_i in frame j, which is the mean of the normal distribution $\bar{x}_j$, is obtained by projecting x_j^{3D} to the image plane. σ is learned from training data.

$P(s_j|x_i, s_i, V, \Delta t)$ models the scale similarity of two detections. We learned a normal distribution from the training data using features $x_i, s_i, V, \Delta t$.

For appearance term $P(a_j|a_i)$, we tried several appearance features and distance measures and finally chose RGB histograms combined with χ^2 distance. The probability distribution is learned by kernel density estimation.

The time gap term $P(\Delta t)$ is defined as [6]:

$$P(\Delta t) = \begin{cases} \mathrm{Z}_t \alpha^{\Delta t - 1}, & 1 \leq \Delta t \leq \xi \\ 0, & \Delta t < 1 \text{ or } \Delta t > \xi \end{cases}$$

where ξ is the maximal allowed time gap, while α is the missing rate of the detector.

2.3. Trajectory Entering and Exit Probability

Since the camera motion only affects similarity model between two detection responses, the unary terms P_{enter} and P_{exit} remains independent to V. P_{enter} is formulated in a posterior way:

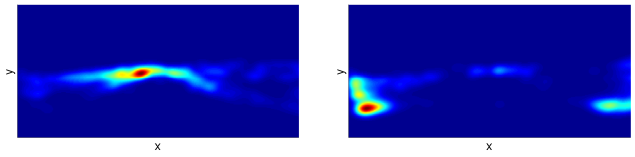
$$P_{enter}(\mathrm{x}_i) = P(start|\mathrm{x}_i) = \frac{P(\mathrm{x}_i|start)P(start)}{P(\mathrm{x}_i)} \quad (5)$$

where $P(\mathrm{x}_i|start)$ stands for the probability density distribution of object entering positions of trajectories. To make things simple, we assume $P(\mathrm{x}_i)$ to be uniform. $P(start)$ is estimated by an EM algorithm described in [6]. $P(\mathrm{x}_i|start)$ is trained by kernel density estimation from training data. $P_{exit}(\mathrm{x}_i)$ is formulated similarly. Fig. 1 shows the learned $P(\mathrm{x}_i|start)$ and $P(\mathrm{x}_i|end)$ from KITTI dataset.

2.4. Detection Probability

$P(\mathrm{x}_i|\mathcal{T})$ represents the likelihood of detection x_i given $\mathcal{T}$. Here we measure it by considering the object scale prior in 3D space. Specifically we first estimate the height of objects in 3D space according to s_i and camera calibration, then compare it to our learned prior model on object heights in 3D and therefor get the likelihood $P(x_i|\mathcal{T})$.

With the above formulation, we solve Eqn. 1 by network flow algorithm as [6].



(a) (b)

Fig. 1. (a)$P(\mathrm{x}_i|start)$ and (b)$P(\mathrm{x}_i|end)$ learned from KITTI dataset. Red implies high probability and blue denotes low probability.

3. EXPERIMENTS

3.1. Setup

For our method focus on first-person multi-object tracking, we use KITTI dataset [14] for experiments. KITTI provides multiple datasets for a variety of tasks related to traffic scene. We choose the dataset provided for tracking that contains 21 sequences with labels that are publicly available. In the experiment, we only consider car tracking. So we remove sequences with very few cars. Since this paper focuses on incorporating the camera motion, we further remove the sequences where the ego car stays static from beginning to end. Finally we picked 16 sequences as our dataset. The training and testing are defined randomly containing 10 and 6 sequences respectively. We use the algorithm mentioned in [13] to estimate the camera motion. We use the outputs of DPMv4 detector pre-trained by KITTI with threshold of -0.5 as our input detection. The maximal allowed time gap ξ is setup as 5.

3.2. Effect of P_{link}

To evaluate the effect of the first three terms of P_{link} in Eqn. 3 which correspond to position, scale and appearance, we use the Bayes classification error. With the label in the ground truth, every pair of bounding boxes in adjacent frames can be divided into 'true pair' where the boxes share same trajectory ID and 'false pair' group where the boxes are labeled with different trajectory IDs. For each term, we calculate the results of the two groups respectively, and compute the Bayes classification error. Table 1 shows the Bayes classification error of the three terms respectively. We can see that our newly defined pairwise terms can improve the classification accuracy which will be useful for tracking.

Table 1. Comparison of Bayes classification error of scale, position and appearance probability of our method and [6].

	[6]	Ours
Position	0.0254	0.0126
Scale	0.0646	0.0595
Appearance	0.3314	0.0689

3.3. Tracking Results

We choose the algorithm [6] as our baseline but without the module to handle occlusion, because we only want to demonstrate the performance of our method with the new terms including transition probability, entering/exit probability, and detection probability. In fact, to show the effect of different terms, we implement several different versions of our method and compare their results as shown in Table 2. We evaluate the tracking performance according to the metrics proposed in [16, 17]. Fully occluded objects and small objects with height less than 25 pixels are ignored during evaluation. Because the parameters may influence the tracking performance, for easy comparison, we evaluate different versions of our method with approximately same recall around 60%. In Table 2, Method 1 is the baseline, Method 2 is the baseline with the new appearance measure. Method 3 to 5 are different versions of improvement based on Method 2. Method 6 is Method 2 plus all the new terms. Method 7 is the Hungarian algorithm with similarity model based on bounding box overlap and our appearance distance. Method 8 is the algorithm proposed by [4]. It can be seen that Method 6 achieves the best MOTA (Multiple Object Tracking Accuracy) compared to all the other methods and its IDS (ID switch) is also much less than the baseline method. Although Method 8 achieves better IDS than our method, its MOTA is much worse.

Fig. 2 shows the tracking results of our method which is Method 6 in Table 2 and baseline method. The bounding boxes with same color indicate that they belong to the same trajectory. We can see that with our method, with the new detection probability term, false alarm of detections disobeying the scale prior are removed, such as the green bounding boxes in the right column in frame 93, 106, 111, and purple ones in the right column in frame 105, 106, 111. Mismatch error is also reduced, for example, from frame 93 to 105 in the right column, the silver cars color changes from cyan to yellow, the black cars color changes from magenta to gray. However, these errors do not show in our method.

4. CONCLUSION

In this paper, we address the multiple-object tracking problem from the first-person viewpoint. The camera motion makes previous work which focus on tracking in static scene unsuitable. This inspired us to introduce the camera motion into the similarity model of two detection responses. We learn a new similarity model that can handle the change of scale and position of objects under the movement of camera. Besides we make use of a new detection prior and appearance similarity measure. Finally, a network flow framework is adopted to solve the tracking problem. Experimental results show that our work can outperform previous work.

Acknowledgement This work is partially funded by NSFC(61103087,91120004,61210005,61272027).

Table 2. Comparison of tracking results of our methods (different versions) with previous work.

#	Method	MOTA	MOTP	recall	precision	MT	PT	ML	Frag	IDS
1	[6] without EOM	0.0111	0.7388	0.6021	0.6147	0.1634	0.5384	0.2981	142	572
2	1+appearance	0.0094	0.7395	0.6012	0.6150	0.1634	0.5288	0.3077	142	577
3	2+position+scale	0.1129	0.7394	0.6006	0.6218	0.1730	0.6058	0.2212	180	243
4	2+detection	0.1622	0.7427	0.6009	0.7282	0.1442	0.6154	0.2404	129	699
5	2+P_{enter} and P_{exit}	0.0251	0.7403	0.6012	0.6228	0.1346	0.5288	0.3365	133	575
6	2+All	0.2902	0.7427	0.6018	0.7226	0.2212	0.5096	0.2692	140	218
7	HM	0.1594	0.7459	0.6508	0.6375	0.2115	0.7115	0.0769	172	240
8	DCT [4]	-0.0088	0.6865	0.6502	0.5384	0.2404	0.5769	0.1827	23	59

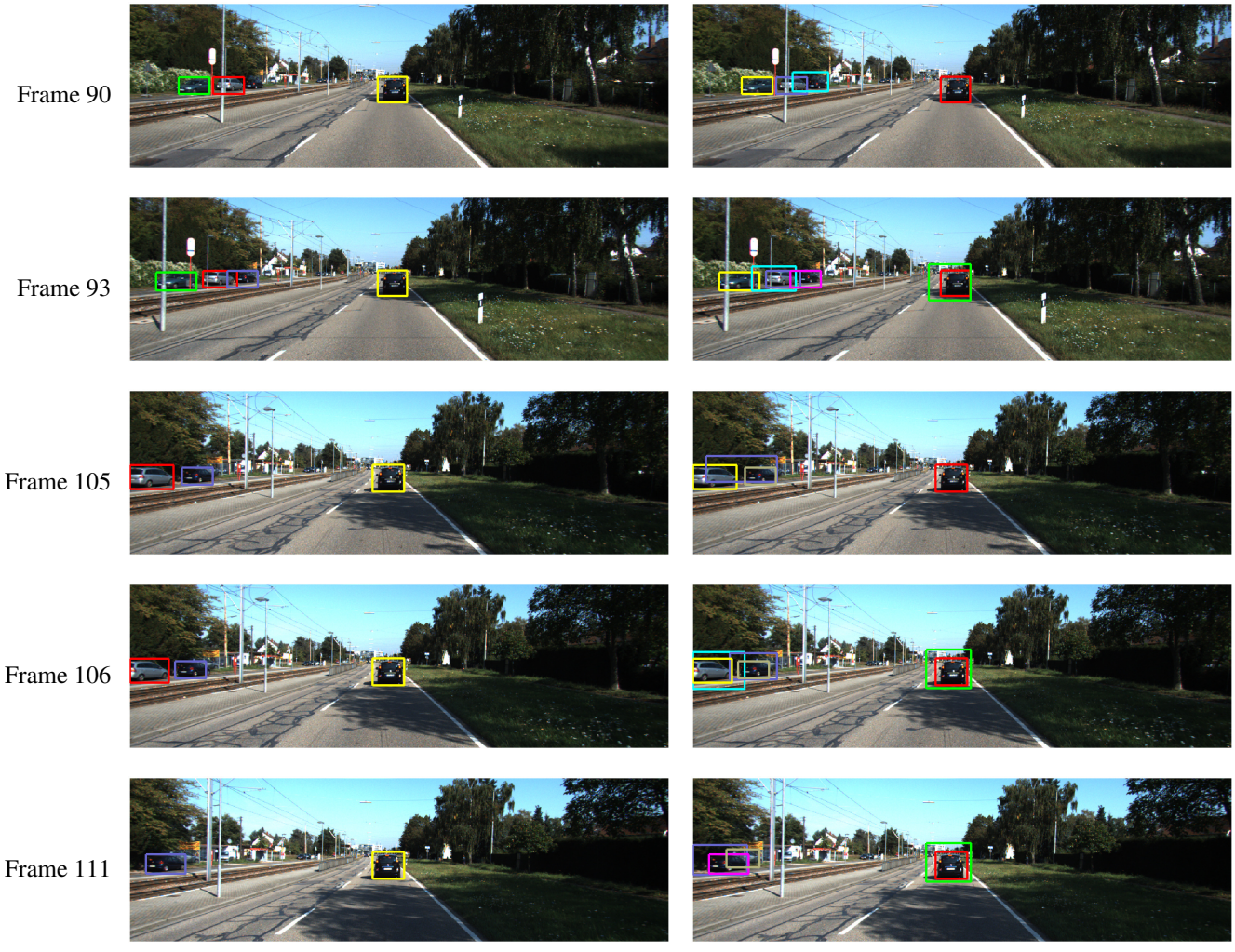


Fig. 2. Sample frames of tracking results of our method and the baseline method.

5. REFERENCES

[1] John MacCormick and Andrew Blake, "A probabilistic exclusion principle for tracking multiple objects," *IJCV*, vol. 39, no. 1, pp. 57–71, Aug. 2000.

[2] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *PAMI*, vol. 32, no. 9, pp. 1627–1645, 2010.

[3] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *CVPR*. IEEE, 2008, pp. 1–8.

[4] Anton Andriyenko, Konrad Schindler, and Stefan Roth, "Discrete-continuous optimization for multi-target tracking," in *CVPR*. IEEE, 2012, pp. 1926–1933.

[5] Cor J Veenman, Marcel JT Reinders, and Eric Backer, "Resolving motion correspondence for densely moving points," *PAMI*, vol. 23, no. 1, pp. 54–72, 2001.

[6] Li Zhang, Yuan Li, and Ramakant Nevatia, "Global data association for multi-object tracking using network flows," in *CVPR*. IEEE, 2008, pp. 1–8.

[7] Hamed Pirsiavash, Deva Ramanan, and Charless C Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *CVPR*. IEEE, 2011, pp. 1201–1208.

[8] Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua, "Multiple object tracking using k-shortest paths optimization," *PAMI*, vol. 33, no. 9, pp. 1806–1819, 2011.

[9] Asad A Butt and Robert T Collins, "Multi-target tracking by Lagrangian relaxation to min-cost network flow," in *CVPR*. IEEE, 2013, pp. 1846–1853.

[10] Andrew V Goldberg, "An efficient implementation of a scaling minimum-cost flow algorithm," *Journal of algorithms*, vol. 22, no. 1, pp. 1–29, 1997.

[11] Robert T Collins, "Multitarget data association with higher-order motion models," in *CVPR*. IEEE, 2012, pp. 1744–1751.

[12] Chetan Arora and Amir Globerson, "Higher order matching for consistent multiple target tracking," in *ICCV*, December 2013.

[13] Andreas Geiger, Julius Ziegler, and Christoph Stiller, "Stereoscan: Dense 3d reconstruction in real-time," in *Intelligent Vehicles Symposium*, Baden-Baden, Germany, June 2011, pp. 963 – 968.

[14] Andreas Geiger, Philip Lenz, and Raquel Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *CVPR*. IEEE, 2012.

[15] Derek Hoiem, AlexeiA. Efros, and Martial Hebert, "Putting objects in perspective," *IJCV*, vol. 80, no. 1, pp. 3–15, 2008.

[16] Bernardin Keni and Stiefelhagen Rainer, "Evaluating multiple object tracking performance: the clear mot metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, 2008.

[17] Yuan Li, Chang Huang, and Ram Nevatia, "Learning to associate: Hybridboosted multi-target tracker for crowded scene," in *CVPR*. IEEE, 2009, pp. 2953–2960.