ONLINE MULTIPLE OBJECT TRACKING VIA FLOW AND CONVOLUTIONAL FEATURES

Lu Wang[⋆] *Lisheng Xu*[⋆] *Min Young Kim*[†] *Luca Rigazio*[†] *Ming-Hsuan Yang*[‡]

[⋆]Northeastern University, China [†]Panasonic Silicon Valley Laboratory, USA [‡]UC Merced, USA

ABSTRACT

We propose an online multiple object tracking algorithm that exploits optical flow and convolutional features to handle noisy detections as well as frequent occlusion. To achieve robust tracking, we develop a data association method that deals with tracking scenarios of increasing difficulty. For easy scenarios, we use motion affinity to associate detections with objects. For ambiguous situations, we propose to use an appearance model based on convolutional features and correlation filters to complement template matching methods. For difficult cases where objects are under heavy occlusion, we carry out occlusion analysis, which exploits the relationship between targets and occluders to predict potential object locations. To deal with noisy detections, false positives are detected and removed on both raw detection and tracklet levels, while missing and inaccurate detections are recovered or corrected via short-term tracking. Experimental results on two benchmark datasets demonstrate that the proposed online algorithm performs favorably against the state-of-the-art methods.

Index Terms— Multiple object tracking, convolutional features, correlation filter, optical flow, occlusion analysis

1. INTRODUCTION

The goal of Multiple Object Tracking (MOT) is to determine the locations and identities of numerous individuals in a complex scene over time. Although significant progress has been made, it remains a challenging problem due to numerous factors including lighting variation, abrupt movement, frequent occlusion, moving cameras, and background clutters.

With recent advances in object detection [1, 2, 3, 4], numerous MOT methods have been developed based on tracking by associating detection responses. As data association is an essential module for an MOT system, various appearance models have been proposed, e.g., color histograms [5, 6], optical flow [7, 8], and classifiers [9, 10]. Convolutional features in deep neural networks have recently been shown to be effective in numerous computer vision tasks [11]. In this work, we exploit convolutional features and correlation filters for robust appearance representation and data association. The proposed model is effective as it uses hierarchical features and represents multiple detections of an object in a running average manner. Therefore, the affinity can also be used for associating detections after a long temporal duration of occlusion. However, this model is less effective in accounting for appearance variation if it is not properly updated (e.g., due to fast object appearance variation). As such, another affinity measure based on optical flow is developed, which can adapt to object appearance change well, but is less effective when noisy detections and wrong associations occur. Thus, exploiting these affinity metrics help achieve more accurate data association for MOT.

Aside from correct association, it is crucial for MOT methods to handle noisy detections [12]. We propose a method to deal with false alarms and missing or inaccurate detections. Specifically, false positives are detected from both the detection and tracklet levels by considering the detection scale, position, score, mutual overlap and tracklet length. To deal with missing or inaccurate detections, we exploit temporal information with a detection-by-tracking scheme.

In MOT, occlusion often causes missing detections and fragmented tracklets. Although this issue may be alleviated by filling temporal gaps between objects and detections, it is hard to predict target locations after occlusion when the camera or object undergos nonlinear motion. For videos with fixed cameras, this problem may be addressed by learning the nonlinear motion pattern of objects to associate tracklets before and after occlusion [10]. Nevertheless, it requires off-line training with a sufficient amount of image data. In contrast, we use the occluder locations to predict the motion of occluded objects and compute affinities based on the convolutional features for robust data association.

The modules mentioned above are effectively integrated into an online MOT algorithm as shown in Figure 1.

2. RELATED WORK

We discuss the related work in terms of two basic modules in MOT methods: appearance model and data association.

Appearance model. Conventional appearance models based on color histograms are not effective for differentiating different objects under occlusion. Discriminative appearance models based on various features of specific objects have been proposed [9, 10]. In addition, numerous metric learning methods have been used to combine features for data association [13, 14]. Recently, an effective optical flow based model for MOT is proposed [7]. Different from existing models that

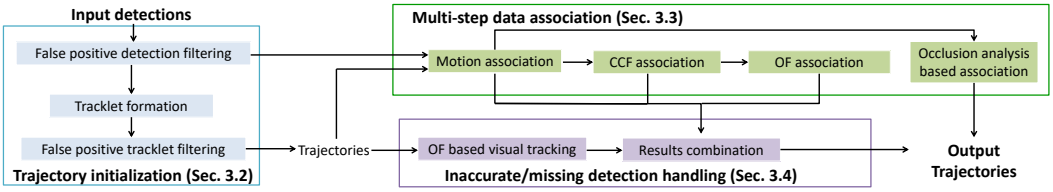

Fig. 1. Main steps of the proposed online multiple object tracking algorithm.

only use static object appearance, a person-specific dynamic scheme with an incremental update is developed to match pedestrians [15]. Siamese convolutional neural networks have recently been used for appearance representation and metric learning for MOT [16, 17, 18], demonstrating the effectiveness of convolutional features for MOT. In this work, we use the highly efficient correlation filter with convolutional features [19] for effective data association.

Data association. For online MOT, data association is typically solved by the Hungarian algorithm [20]. Network flow based methods have also been adopted for online tracking [21]. In [8] the online MOT task is formulated as a Markov decision process and a reinforcement learning method is applied to associate objects and detections. To make the association process efficient and reliable, [22] solves associations from low to high cost to deal with ambiguous associations gradually. Recently, [23] and [18] employ long-short term memory for association. Our proposed data association method uses motion, appearance and occlusion analysis progressively to determine the association of objects and detections with increasing difficulty. Furthermore, in this work data association is coupled with a detection-by-tracking scheme based on optical flow to recover missing and rectify inaccurate detections.

3. PROPOSED ONLINE MOT ALGORITHM

We first introduce the appearance, motion and scale affinity metrics used in data association, and then present the proposed online MOT algorithm in terms of three modules: trajectory initialization, data association, and detection-by-tracking for missing or inaccurate detection of objects.

We denote an object by o_i and its state at frame t by o_i^t, while a detection from frame t is represented by z_j^t.

3.1. Affinity Metrics

Appearance affinities. To measure appearance affinity, image patches of detections are normalized to the same size.

To calculate the optical flow affinity A_{OF}, given two detections, each normalized image patch is considered as a template from which a set of points are densely and uniformly sampled and then matched to the other image patch using the forward-backward optical flow [24] based on the iterative Lucas-Kanade method with pyramids [25]. A sample point is not matched if the end of its corresponding backward flow deviates significantly (larger than 10 pixels in this work). To increase the discriminability, the flow matching score is rectified by setting the value below a threshold (0.985 in this work) to be 0 and re-scaling the remaining non-zero part to (0,1]. Next, A_{OF} is computed as the average matching score of all sampled points.

To compute the convolutional correlation filter (CCF) [19] based affinity A_{CCF}, for a trajectory that is newly initialized, the CCF representing the object is learned from the most confident Q detections of the trajectory ($Q = 2$ in this work). Then the learned CCF is applied to the detection to be associated to generate a response map f. The maximum value around the center of f (one-third of the target region's side length in this work) measures A_{CCF} between the object and the detection. Due to space limitation, readers are referred to [19] on how to calculate a CCF and the response map.

Motion affinity. The Kalman filter is applied to predict the target position in the following frames. The motion affinity between an object $o_i^{t_1}$ and a detection $z_j^{t_2}$ is defined to be the intersection-over-union (IOU) ratio between the predicted bounding box of $o_i^{t_1}$ at t_2 and the bounding box of $z_j^{t_2}$.

Scale affinity. The scale affinity between two detections is computed by $1 - 0.5 \cdot |h_i - h_j| / (h_i + h_j) - 0.5 \cdot |w_i - w_j| / (w_i + w_j)$, where h and w represent the height and width of the detection, respectively. The scale affinity between an object o_i and a detection z_j is computed as the scale affinity between last associated detection of o_i and z_j.

3.2. Trajectory Initialization

False positive detection removal. Before trajectory initialization, detections are classified as either positive or negative by a linear SVM classifier, learned from the training images of objects, according to the detection's scale, position, and the detection score.

Another source of false positives is multiple detections of one object instance, which may result in redundant trajectories and fragmentation. Therefore, to differentiate if two overlapped detections correspond to one object or two, another linear binary classifier is learned by using the IOU, scale difference and detection score difference of overlapped detection pairs as features.

Tracklet formation and trajectory initialization. After removing false positives, remaining detections are used to form object trajectories. Detections from adjacent frames that are not associated with any existing objects are linked by using distance and the optical flow appearance affinity A_{OF}. Two

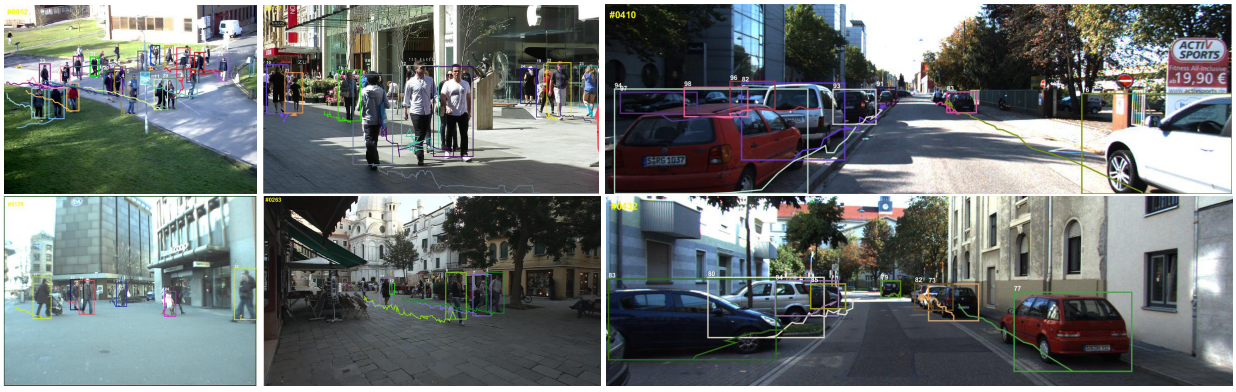

Fig. 2. Sample tracking results on the MOT 2015 (left two columns) and KITTI (right column) datasets.

detections are linked when their image distance is small and there is no other possible conflict link. Otherwise, the distance constraint is relaxed and A_{OF} is computed to determine if the two detections should be connected. A_{CCF} is not used here as, according to our experiment, for adjacent frame association, A_{OF} has similar performance to A_{CCF} but requires much less computational cost than A_{CCF}.

A new object is initialized only when the length of the tracklet is above the threshold T_{init} and it is classified as a true target. As tracklets correspond to false positives are mostly with low detection scores and short, the classifier for differentiating true objects from false ones takes the mean detection score and length of each tracklet as the features. .

3.3. Data Association

The trajectories up to time $t-1$ and detections at time t are connected based on affinity measures. In this work, motion affinity is first used for association to deal with easy scenarios. An assignment is formed when the motion affinity is high enough, and there is no other conflicting association whose motion affinity is also high.

The computationally more expensive appearance affinities are then computed for only a small percent of object-detection pairs that are not associated but satisfy the motion and scale affinity constraints. A_{CCF} is first computed and the assignment is solved using the Hungarian algorithm. An assignment is not accepted if the corresponding A_{CCF} is lower than a threshold. For sequences with low frame rate and acquired with moving platforms, the appearance of the same object may change quickly. In such scenarios, A_{OF} is used complimentarily to find qualified association pairs.

Occlusion analysis is finally used to infer missing trajectories. As linear motion prediction for a longer duration is ineffective especially when the object or the camera is under nonlinear motion, in such cases we use likely occluders to recover the missing trajectory segments of targets. The assumption is that an occluded object is likely to reappear near its occluder. When the tracker loses track of an object, the potential occluders are recorded. Then, after a temporal duration when the occlusion process is likely to finish, if o_i is still not being tracked, the occlusion reasoning module is activated. The possible association to o_i is searched at the locations around those potential occluders. If a match based on A_{CCF} is satisfied by a detection z_j, z_j is associated with o_i. The missing part of the trajectory for o_i is then inferred by linearly interpolating the relative location of o_i to its occluder before and after occlusion to account for nonlinear motion.

The trajectory of an object is terminated if the object is not associated with any detection for a certain duration (e.g., 1 second in this work) or the object exits the field of view.

3.4. Handling Missing and Inaccurate Detections

After the data association at time t, we recover missing detections for objects that are not associated with detections, and rectify inaccurate detections for objects that are not reliably associated. To this end, if an object o_i has associated with a detection z_j^{t-1} at $t-1$, we compute the optical flow of z_j^{t-1} to estimate the bounding box bb_j^t of o_i at t for visual tracking.

When o_i is associated with a detection z_k^t but with low motion and appearance affinity values, z_k^t is compared with bb_j^t. If the IOU is small (e.g., less than 0.5) or their scales are not similar ($A_{scale} < 0.9$), they are considered inconsistent. Then the affinity value $A_{CCF}(o_i, bb_j^t)$ and $A_{CCF}(o_i, z_k^t)$ are computed. The one with the higher A_{CCF} is selected for association.

When o_i is not associated with any detection, the visual tracking result bb_j^t is associated with o_i if the detection associated in the last frame is reliable (i.e., with a high detection score).

With this approach, the association in following frames can also be more accurate as the corresponding motion and appearance affinities can be more accurately computed.

3.5. Online Update of CCF

In this work, instead of using a fixed update rate [19], only when the motion affinity is high (e.g., ≥ 0.7) while the appearance affinity is not low (e.g., ≥ 0.5), the associated de-

Table 1. Performance of the proposed algorithm on the MOT 2015 training data after disabling different modules. Meanings of abbreviations: FPRe - false positive removal, FTRe - false tracklet removal, OA - occlusion analysis, DRec - missing/inaccurate detection rectify, CFUd - CCF update.

Disabled module	FPRe	FTRe	OA	DRec	CCF	CFUd	None
MOTA	33.3	35.3	36.5	34.6	37.4	38.3	39.8

Table 2. Tracking results on the MOT 2015 dataset (based on the ACF [2] detections provided by the website).

	Method	MOTA	MOTP	MT	ML	IDs	Frag
Online	RMOT [6]	18.6	69.6	5.3	53.3	684	1282
	SCEA [29]	29.1	71.1	8.9	47.3	604	1182
	MDP [8]	30.3	71.3	13.0	38.4	680	1500
	Proposed	31.6	71.8	10.1	46.3	491	994
Batch	LINF1 [30]	24.5	71.3	5.5	64.6	298	744
	LP_SSVM [31]	25.2	71.7	5.8	53.0	646	849
	SiameseCNN [17]	29.0	71.2	8.5	48.4	639	1,316
	CNNTCM [16]	29.6	71.8	11.2	44.0	712	943
	MHT_DAM [32]	32.4	71.8	16.0	43.8	435	826
	NOMT [7]	33.7	71.9	12.2	44.0	442	823

tection is considered reliable and used to update CCF. For objects that are being occluded, CCF is not updated. The learning rate is set to 0.01. Figure 2 shows sample tracking results of the proposed algorithm on two datasets.

4. EXPERIMENTAL RESULTS

4.1. Experimental Setups

We evaluate the proposed algorithm against the state-of-the-art methods on the MOT 2015 [26] dataset for pedestrian tracking and KITTI [27] dataset for car tracking.

The parameter T_{init} is empirically set to $\max(5, FPS/3)$ for both datasets. Other parameters and thresholds are learned from the training sequences. The motion affinity thresholds are 0.65 for pedestrians and 0.55 for cars respectively, while the scale affinity thresholds are 0.85 for pedestrians and 0.7 for cars. These two thresholds are set differently for different types of objects because cars' aspect ratio varies much more significantly than pedestrians. The appearance affinity thresholds for CCF and optical flow are 0.35 and 0.5 respectively. These parameters are fixed in all experiments.

We implement the proposed method in MATLAB and the MatConvNet [28] toolbox to extract convolutional features on a computer with Core i7 3.6G Hz CPU, 16G memory, and NVIDIA GeForce GT 720 GPU. The proposed method runs at 0.9 FPS. Without using CCF, our method runs at 5 FPS. More results are available at `http://faculty.neu.edu.cn/ise/wanglu/CCF_MOT.htm`.

4.2. Contributions of Different Modules

To demonstrate the contribution of each module of our approach, we evaluate the performance of our approach on the MOT training data by removing one module at a time. We divide the samples into two sets same as [8], one for training and the other for validation, and vice versa.

Table 3. Car tracking results on the KITTI dataset (based on the regionlet [3] detections provided by the website).

	Method	MOTA	MOTP	MT	ML	IDs	Frag
Online	RMOT [6]	53.0	75.4	39.5	10.1	215	742
	mbodSSP [21]	62.6	78.8	48.0	8.7	116	884
	SCEA [29]	67.1	79.4	52.1	11.0	106	466
	NOMT-HM [7]	67.9	80.0	49.2	13.1	109	371
	Proposed	69.5	78.4	52.3	13.1	72	408
Batch	LP_SSVM [31]	66.4	77.8	56.0	8.2	63	558
	NOMT [7]	69.7	79.5	56.3	13.0	36	225

The results on the validation sets are listed in Table 1. We draw the following observations based on the results shown in Table 1: 1). Without the false positive removal, the MOTA drops as much as 6.5%, which demonstrates the importance of dealing with false positives. 2). Without analyzing false positive tracklets, the MOTA also drops a lot (4.5%). 3). Occlusion analysis based association helps filling temporal gaps more accurately and improves the overall tracking result. 4). Dealing with missing/inacurate detections through visual tracking increases the recall rate of the tracker. 5). Replacing CCF with optical flow, the MOTA drops for 2.4%. 6). Proper CCF update strategy is important for improving the performance of the tracker.

4.3. Evaluation on the MOT and KITTI Datasets

We submit the tracking results of the proposed method on the MOT 2015 and KITTI test data to the respective websites and obtain the evaluation results as shown in Table 2 and 3. For both pedestrian and car tracking, the proposed algorithm performs favorably against the state-of-the-art methods, especially the online MOT methods. From the tables we can further see that our MOT tracker achieves the smallest number of identity switches among all the online trackers and produces smaller number of trajectory fragmentations as well, which are mainly attributed to the application of CCF and optical flow as the appearance model and the explicit occlusion analysis for occlusion handling.

5. CONCLUSIONS

In this paper, we propose an online multiple object tracking algorithm. The proposed data association scheme uses motion, appearance affinities and occlusion analysis to solve object-detection assignments for different cases of increasing difficulty. To ensure reliable association, a convolutional correlation filter is adopted to compliment the template matching module based on optical flow. The CCF is also used to analyze visual tracking results to handle missing and inaccurate detections, as well as for robust association in occlusion analysis. Experimental results on two challenging benchmark datasets show that the proposed online algorithm performs favorably against the state-of-the-art methods.

Acknowledgement Lu Wang is supported in part by Chinese Scholarship Council, in part by NSFC #61202258, and in part by NSF of Liaoning, China #20170540312 .

6. REFERENCES

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

[2] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *PAMI*, vol. 36, no. 8, pp. 1532–1545, 2014.

[3] X. Wang, M. Yang, S. Zhu, and Y. Lin, "Regionlets for generic object detection," *PAMI*, vol. 37, no. 10, pp. 2071–2084, 2015.

[4] R. Girshick, "Fast R-CNN," in *ICCV*, 2015, pp. 1440–1448.

[5] C. Huang, B. Wu, and R. Nevatia, "Robust object tracking by hierarchical association of detection responses," in *ECCV*, 2008, pp. 788–801.

[6] J.H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon, "Bayesian multi-object tracking using motion context from multiple objects," in *WACV*, 2015, pp. 33–40.

[7] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *ICCV*, 2015, pp. 3029–3037.

[8] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: online multi-object tracking by decision making," in *ICCV*, 2015, pp. 4705–4713.

[9] C.-H. Kuo and R. Nevatia, "How does person identity recognition help multi-person tracking?," in *CVPR*, 2011, pp. 1217–1224.

[10] B. Yang and R. Nevatia, "Multi-target tracking by online learning of non-linear motion patterns and robust appearance models," in *CVPR*, 2012, pp. 1918–1925.

[11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[12] L. Wen, D. Du, Z. Cai, Z. Lei, M.-C. Chang, H. Qi, J. Lim, M.-H. Yang, and S. Lyu, "Detrac: a new benchmark and protocol for multi-object tracking," *arXiv preprint arXiv:1511.04136*, 2015.

[13] W. Luo, J. Xing, X. Zhang, X. Zhao, and T.-K. Kim, "Multiple object tracking: A literature review," *arXiv preprint arXiv:1409.7618*, 2014.

[14] B. Wang, G. Wang, K.L. Chan, and L. Wang, "Tracklet association by online target-specific metric learning and coherent dynamics estimation," *PAMI*, 2016.

[15] M. Yang and Y. Jia, "Temporal dynamic appearance modeling for online multi-person tracking," *arXiv preprint arXiv:1510.02906*, 2015.

[16] B. Wang, K. L. Chan, L. Wang, B. Shuai, Z. Zuo, T. Liu, and G. Wang, "Joint learning of convolutional neural networks and temporally constrained metrics for tracklet association," in *CVPR DeepVision Workshop*, 2016.

[17] L. Leal-Taixé, C. Canton-Ferrer, and K. Schindler, "Learning by tracking: siamese cnn for robust target association," in *CVPR DeepVision Workshop*, 2016.

[18] A. Sadeghian, A. Alahi, and S. Savarese, "Tracking the untrackable: Learning to track multiple cues with long-term dependencies," *arXiv preprint arXiv:1701.01909*, 2017.

[19] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *ICCV*, 2015, pp. 3074–3082.

[20] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

[21] P. Lenz, A. Geiger, and R. Urtasun, "Followme: Efficient online min-cost flow tracking with bounded memory and computation," in *ICCV*, 2015, pp. 4364–4372.

[22] F. Solera, S. Calderara, and R. Cucchiara, "Learning to divide and conquer for online multi-target tracking," in *ICCV*, 2015, pp. 4373–4381.

[23] A. Milan, S.H. Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," in *AAAI*, 2017.

[24] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *PAMI*, vol. 34, no. 7, pp. 1409–1422, 2012.

[25] J.-Y. Bouguet, "Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm," *Intel Corporation*, vol. 5, pp. 1–10, 2001.

[26] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "Motchallenge 2015: Towards a benchmark for multi-target tracking," *arXiv preprint arXiv:1504.01942*, 2015.

[27] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: the kitti dataset," *IJRR*, 2013.

[28] A. Vedaldi and K. Lenc, "Matconvnet: convolutional neural networks for matlab," in *ACM MM*. ACM, 2015, pp. 689–692.

[29] J.H. Yoon, C.-R. Lee, M.-H. Yang, and K.-J. Yoon, "Online multi-object tracking via structural constraint event aggregation," in *CVPR*, 2016, pp. 1394–1400.

[30] L. Fagot-Bouquet, R. Audigier, Y. Dhome, and F. Lerasle, "Improving multi-frame data association with sparse representations for robust near-online multi-object tracking," in *ECCV*, 2016.

[31] S. Wang and C. Fowlkes, "Learning optimal parameters for multi-target tracking," in *BMVC*, 2015.

[32] C. Kim, F. Li, A. Ciptadi, and J.M. Rehg, "Multiple hypothesis tracking revisited," in *ICCV*, 2015, pp. 4696–4704.