IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Multi-Agent Deep Reinforcement Learning for Multi-Object Tracker

Mingxin Jiang[1,2], Chao Deng[3],Yinshan Yu[1,2], Jingsong Shan[4]

[1]Jiangsu Laboratory of Lake Environment Remote Sensing Technologies, Huaiyin Institute of Technology, Huaian, 223003, China;
[2]Faculty of Electronic information Engineering, Huaiyin Institute of Technology, Huaian, 223003, China;
[3]School of Physics & Electronic Information Engineering, Henan Polytechnic University, Jiaozuo, 454000, China;
[4]Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian, 223003, China.

Corresponding author: Chao Deng (e-mail: dengchao_hpu@163.com) and Yinshan Yu (e-mail: yyshyit@126.com).

This work was supported by Major Program of Natural Science Foundation of the Higher Education Institutions of Jiangsu Province under Grant 18KJA520002, Project funded by the Jiangsu Laboratory of Lake Environment Remote Sensing Technologies under Grant JSLERS-2018-005, Six talent peaks project in Jiangsu Province under Grant 2016XYDXXJS-012, the Natural Science Foundation of Jiangsu Province under Grant BK20171267, the fifth issue 333 high-level talent training project of Jiangsu province (BRA2018333), 533 talents engineering project in Huaian under Grant HAA201738, National Natural Science Foundation of China under Grant(61801188).

ABSTRACT Multi-object tracking has been a key research subjects in many computer vision applications. We propose a novel approach based on multi-agent deep reinforcement learning (MADRL) for multi-object tracking to solve the problems in the existing tracking methods, such as a varying number of targets, non-causal, non-realtime, etc. At first, we choose YOLO V3 to detect the objects included in each frame. And unsuitable candidates were screened out and the rest of detection results are regarded as multiple agents and forming a multi-agent system. Independent Q-Learners (IQL) is used to learn the agents' policy, in which, each agent treats other agents as part of the environment. Then, we conducted offline learning in the training and online learning during the tracking. Our experiments demonstrate that the use of MADRL achieves better performance than the other state-of-art methods in precision, accuracy and robustness.

INDEX TERMS Multi-object tracking; MADRL; IQL; YOLO V3

I. INTRODUCTION

Visual multi-object tracking is one of the crucial problems in computer vision field and has a wide range of applications, such as, robotics, artificial intelligence, virtual reality and so on [1-4]. Despite great successes in the last decades, multi-object tracking still remains challenging due to a lot of factors including object appearing or disappearing, occlusion, appearance similarity, background clutter [5-7].

In recent progress on multi-object tracking, tracking-by-detection strategy has been focused on due to rapid development for object detection methods [8-11]. To overcome ambiguities in associating object detections and resolve the detection failures, some research papers take future time steps into account, which are not suitable for online tracking applications, for example, autonomous driving and robot navigation because they are not causal systems [12-15].

In most of recent works, tracking-by-detection multi-object tracking approaches are roughly divided into two categories: offline mode and online mode [16-18]. In offline learning, we can perform learning before the actual tracking happens, in which the detections of all the frames in the video sequence are often used together to avoid detection failures. The offline learning use ground truth of objects' trajectories to complete supervised learning which can prevent tracking drift happening. A cluttered or crowded scene usually brings some difficulties as the offline learning is static and cannot consider the dynamic of the object in the history of data association. To overcome these difficulties, the global data association is used in many multi-object tracking algorithms. However, only using the offline approaches, the tracking performance is still limited and it is hard to be applied to real-time applications.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901300, IEEE Access

On the contrary, online methods perform learning during tracking which can be applied to real-time applications [19-20]. The major challenge is the ambiguities in associating noisy detections in the current frame with the tracked objects in the previous frame. To handle this challenge, different cues, such as motion and appearance, often are combined in association. The hand-crafted features, such as Harr-like features [21], histograms of oriented gradients (HOG)[22], and local binary patterns (LBP)[23] are used frequently in most previous multi-object tracking methods. As more complex characteristics of the objects cannot be captured using hand-crafted features, these exiting methods have many limits in applications. In addition, ground truth is not taken into account for supervision in the online learning, some incorrect training examples may lead to tracking drift.

Reinforcement learning (RL) has gained some success in the previous researches, but these existing approaches have poor scalability and are limited to low-dimensional issues [24-26]. There are these limitations mainly because RL has higher complexity. With the rising of deep learning, new solutions have been provided to solve these problems. As deep neural networks can provide powerful function approximation and deep feature representations, deep reinforcement learning (DRL) can perform more effective than RL. Compact low-dimensional features of high-dimensional data (such as images, text, and audio) can be found by deep neural networks automatically, which is the most outstanding contribution of deep learning. In the last few years, DRL has achieved rapid progress and opened entrances to a new perspective on this issue[27-29], and has been applied in many emerging domains [30-35].

Generally speaking, DRL considers a single agent in a stationary environment, namely single agent deep reinforcement learning (SADRL). By comparison, multi-agent deep reinforcement learning (MADRL) takes multiple agents learning into account and has received an increased amount of attention [36-38], but rarely is applied in visual multi-object tracking. Unlike SADRL, converge often fails in MADRL because the objects always move. In MADRL setting, multiple agents' rewards is related to each agent's actions, and finding optimal policies become difficult.

Based on the above analysis, we propose a multi-object tracking algorithm by using MADRL, which can improve the performances in both precision and accuracy. Figure 1 illustrates the pipeline of our proposed tracker, the important contributions can be summarized as follows:

- A tracker based on MADRL is proposed to solve the problems in the existing tracking methods, such as a varying number of targets, non-causal, non-realtime, etc. To the best of our knowledge, we are the first to apply MADRL to solve the problem of visual multi-object tracking.
- In our tracker, YOLO V3 is adopted as object detector as it has state-of-art performance and is a real-time detection system. A single frame image is considered as an environment, each single object is formulated as an agent, a set of agents in the shared environment forms a multi-agent system. IQL is used as it is more practical in processing multi-object tracking problem, in which, each agent learns its own policy independently, and treats other agents as part of the environment.
- Learning a similarity function for data association in multi-object tracking is equivalent to learning a policy in MADRL. We conducted offline learning in the period of training and online learning during the tracking phase, which take full advantage of offline learning and online learning.

The rest of our paper is structured as follows: the background is reviewed in the following section. Section III. introduces our proposed multi-object tracking method. The experimental results and analysis are demonstrated in Section IV. Finally, we draw conclusions in Section V.

II. BACKGROUND

A. SINGLE-AGENT DEEP REINFORCEMENT LEARNING (SADRL)

A traditional RL problem can be described as a Markov decision process (MDP), in which the agent aims to make a sequence decisions. RL provides a coherent framework, an agent can learn from an environment a policy function that maps states to actions and take actions in order to maximize its expected cumulative rewards at each discrete time step.

Formally, RL defines an environment $\mathcal{E}$, and the state $s \in \mathcal{S}$ of an agent at time step t, the agent need to perform an action $u \in U$, and a reward function R can help the agent to learn an optimal policy $\pi(a|s)$ to choose an action based on its states. A state transition function $\mathrm{P}(s'|s,a)$ which map a pair of state-action at time step t onto a distribution of states at time step t+1.

The goal of the agent is to maximize its expected cumulative rewards $R=\sum_{t=0}^{\infty}\gamma^t r_t$, where $\gamma \in [0,1)$ is the discount factor and r_t is a reward signal that the agent receives from the environment at time step t during the training process. In tracking method, reward r_t is given at the end of a tracking episode when the object is tracked successfully. More specifically, the reward signal $r_t=0$ during iteration at each time step. When 'stop' action is selected at termination step T, the reward signal r_T is a thresholding function of IoU as follows:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901300, IEEE Access

$$r_T = \begin{cases} 1 & if \quad IoU(p_T, \mathrm{g}) > \tau \\ -1 & otherwise \end{cases} \tag{1}$$

where $IoU(p_T, \mathrm{g}) = \mathrm{area}(p_T \cap g) / \mathrm{area}(p_T \cup g)$ represents overlap ratio of p_T and the ground truth of the object.

B. MULTI-AGENT DEEP REINFORCEMENT LEARNING (MADRL)

Different from SADRL, MADRL considers multiple agents learning by RL and the non-stationarity caused by other agents changing their behaviors when they learn. A set of agents in a shared environment, which must learn to maximize their individual returns, are involved in MADRL.

Deep Q-Network (DQN) is one of popular methods that used to find an optimal action-selection policy in DRL algorithms. DQN is a form of Q-learning with function approximation using a neural network, which means it tries to learn a state-action value function Q given by a neural network in DQN by minimizing temporal-difference errors. A recurrent neural network parameterized by θ is usually used to represent the Q-function in deep Q-learning. The action-value function Q of a policy π is:

$$Q^{\pi}(s,a|\theta) = \mathbb{E}[R_t | s_t = s, a_t = a, \pi] \tag{2}$$

Given $Q^{\pi}(s,a|\theta)$, the best policy can be found by

$$Q^{*}(s,a|\theta) = \arg\max_{u} Q^{\pi}(s,a|\theta) \tag{3}$$

The function is defined as the Bellman equation[39] to learn Q^{π} actually, which has the following recursive form,

$$Q^{*}(s,a|\theta) = \mathbb{E}_{s'}[r + \gamma Q^{*}(s',a')|\theta] \tag{4}$$

The agent can choose actions at each time step on the basis of the exploration policy, e.g. an $\varepsilon-$ greedy policy can take the currently estimated best action with probability $1-\varepsilon$, and selects a random exploratory action with probability ε. At each iteration i, experience tuple $\langle s,a,r,s' \rangle$ is stored in a reply memory $\mathbf{M}$ and the parameters of DQN θ are updated to minimize the loss function,

$$L(s,a|\theta^i) = \sum_{i=1}^{n} [(r^i + \gamma \max_{a'} Q(s',a'|\hat{\theta}^i) - Q(s,a|\theta^i))^2] \tag{5}$$

The parameters of target network are in combination with experience reply and updated less frequently, that are important for stable deep Q-learning.

III. PROPOSED METHOD

A brief architecture of our proposed multi-object tracking algorithm based on MADRL will be shown in the following subsections firstly. And we will describe the details of our method in the rest of this paper.

A. PIPELINE OF OUR ALGORITHM

The pipeline of our method is demonstrated in Figure 1. Firstly, multiple objects are detected by YOLO V3 [40], which is a state-of-the-art, real-time object detection system. In each frame, YOLO V3 is applied and will output a set of results of detection D_t at time step t, which may include different kinds of objects. We compute the intersecton-over-union (IoU) distance between the ground truth and the results of detection at first frame to get the detections to the tracked. Then, the selected results of object detection are considered as multiple agents and forming a multi-agent system. At last, we adopt a MADRL that can learn to obtain a joint action for multiple objects and get the multi-object tracking results.

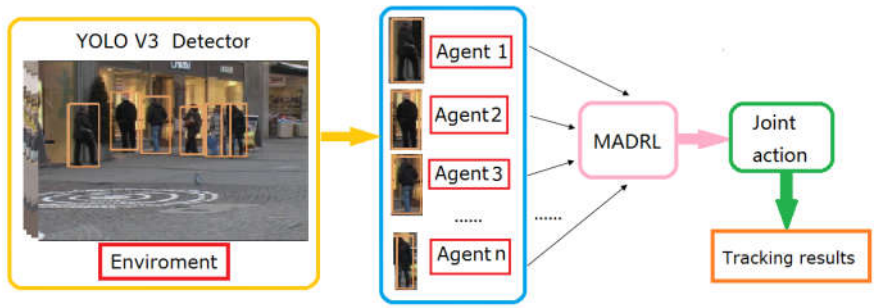


FIGURE 1. Pipeline of multi-object tracking algorithm based on MADRL

B. MULTI-OBJECT TRACKER VIA MADPL

The problem of multi-object tracking is solved by a MADRL method in our tracker. Our MADRL framework is shown as Figure 2, and details of these components will be presented in this section.

In our formulation, we consider a single frame image as an environment. In a multi-agent setting n agents are described by $i \in I \equiv \{1,...,\mathrm{n}\}$, each agent takes a set of actions,forming a joint action $a \in \mathbf{A} \equiv A^n$,to achieve its goal, the agent 's information of the current environment is represented by a set of states $s \in \mathcal{S}$, state transition

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901300, IEEE Access

probabilities are defined by $P(s'|s,a)$. Each agent's observations $z \in Z$ are governed by an observation function $O(s,a)$. The i th agent i selects its action based on its own action-observation history $\tau_i \in T$ according to its policy $\pi(a^i|\tau^i)$. After each state transiton, the new observation $O(s,u)$ and the action a^i are added to τ^i, then $\tau^{i'}$ come into being.

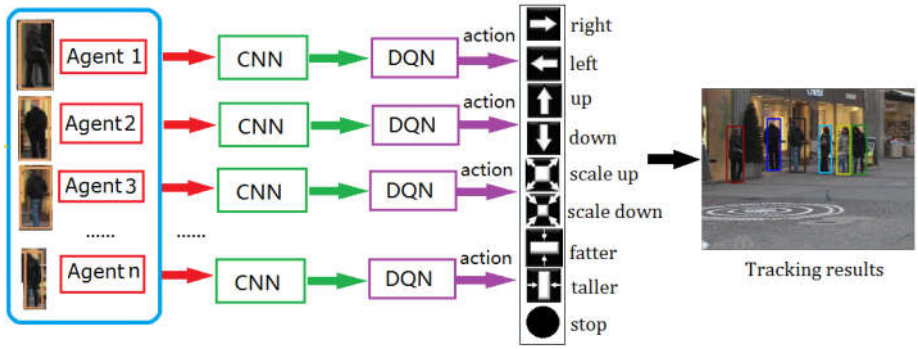


FIGURE 2. **Flow chart of the MADRL**

In our method, we adopt the deep Q-learning algorithm, and the details of the proposed DQN are illustrated as Figure 3. The input agent is processed by a pre-trained CNN, which is conducted on the VGG-16 network and includes five pooling stages and one fully connected layer, i.e. Conv1-2, Conv2-2, Conv3-3, Conv4-3, Conv5-3. The output of the CNN is the state representation of the agent, which is concatenated with the action history. Then, it is input into the DQN which can output the prediction of the value of the actions. The value of actions are applied to the bounding box which is composed of eight actions and one action to terminate the tracking process. Each action is encoded by the 9-dimensional vector, which are defined as follows: {move right, move left, move up, move down, scale up, scale down, aspect ratio change fatter, aspect ratio change taller, stop}. Motivated by [41], we adopt LSTM cells to exchange messages among the agents.

Suppose the Q-network function of the i th agent is $Q(s^i, a^i|\theta_a^i)$, the inter-agent communication is $Q(s^i, a^i, m^i, m^{-i}|\theta_a^i, \theta_m^i)$, where m^i represents the messages that sent out from agent i, and m^{-i} the messages that agent i received from other agents. The message is formalized as a function $m(s,a|\theta_m)$, where θ_m is learned by using deep learning method, which is outperform handcrafted features.

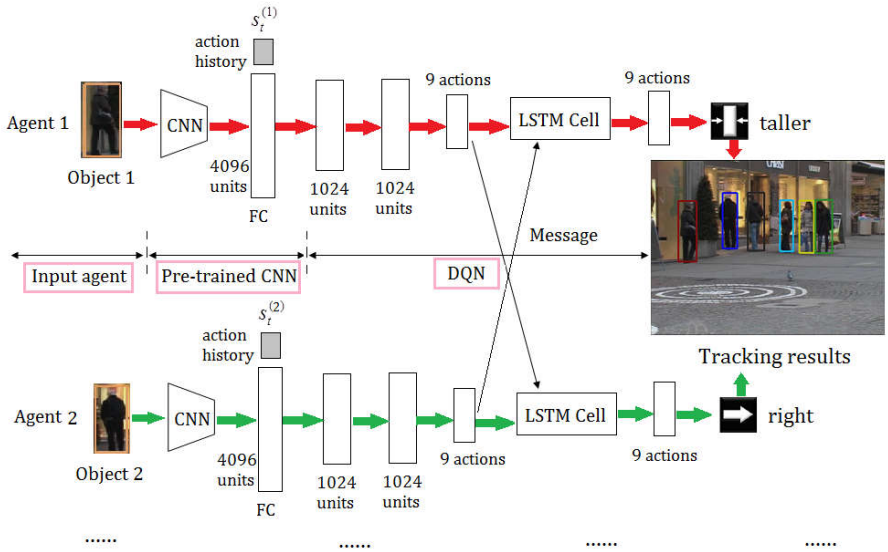


FIGURE 3. **The details of the proposed DQN**

C.MULTI-AGENT IMPORTANCE SAMPLING AND LEARNING

MADRL falls into two major categories: Independent Q-Learners(IQL) and Joint Action Learners(JAL). Only local

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901300, IEEE Access

actions are observed by the agent in IQL, and actions taken by all agents are observed in JAL. IQL is utilized in our approach as it is more practical in processing multi-object tracking problem. In IQL, each agent learns its own policy independently, and treats other agents as part of the environment. However, IQL introduces an important problem: the environment becomes non-stationary from agents' local perspectives due to multiple agents' the interactions with the environment. Each agent has to coordinate with fellow agents so that MADRL has higher effectiveness.

In our method, the non-stationary that caused by IQL is addressed by adopting an importance sampling scheme for the multi-agent setting. In MADRL, we sample the action a_t^i of agent i at time step t, and sample all the agents, according to both the messages sent out from itself and from other agents. According to Eq.5, we can find that the goal of updating the parameters of the Q-network is to minimize the following importance loss function for agent i :

$$L(s_t^i, a_t^i \left| \theta_a^i, \theta_m^i \right.) = \sum_{i=1}^{n} [(r_t^i + \gamma \max_{a''} Q(s_{t+1}^i, a'' \left| \hat{\theta}_a^i, \hat{\theta}_m^i \right.) - Q(s_t^i, a_t^i, m_t^i, m_t^{-i} \left| \theta_a^i, \theta_m^i \right.))^2] \tag{6}$$

Learning a similarity function for data association in multi-object tracking is equivalent to learning a policy in MADRL. Motivated by [42], we conducted offline learning in the period of training and online-learning during the tracking phase, because the ground truth can be used for supervision to avoid the tracking drift in offline learning, at the same time, the dynamic status and the history of the target object can be taken into account in online-learning.

IV. EXPERIMENTS

A.IMPLEMENTATION DETAILS

The experiments of our proposed multi-object tracking algorithm were conducted on a workstation equipped with the Windows 10 operating system, Intel(R) Core(TM) i7-4712MQ CPU, 32GB RAM, and GeForce GTX TITAN X GPU, 12.00 GB VRAM. We used MATLAB R2016b as our software platform. In CNN, the learning rate is set to 0.0001 for convolutional layers and is set to 0.001 for fully-connected layers.

B. QUANTITATIVE EVALUATION

In this section, our approach (MADRL) is compared with other five state-of-art multi-object trackers, i.e. MDPSubCNN[42], RNN-LSTM[43], SiameseCNN[44], LP_SSVM[45], LSTM_DRL[46], on the MOT challenge benchmark[47] in order to evaluate the tracking performance. We use the CLEAR MOT metrics for quantitative evaluation including the multiple object tracking accuracy (MOTA), the multiple object tracking precision (MOTP). On the 8 test videos that have public CLEAR MOT metrics data included in the MOT Challenge dataset, the quantitative comparison is conducted between our tracker with other state-of-art trackers, and the results are reported in Figure 4 and Figure 5.

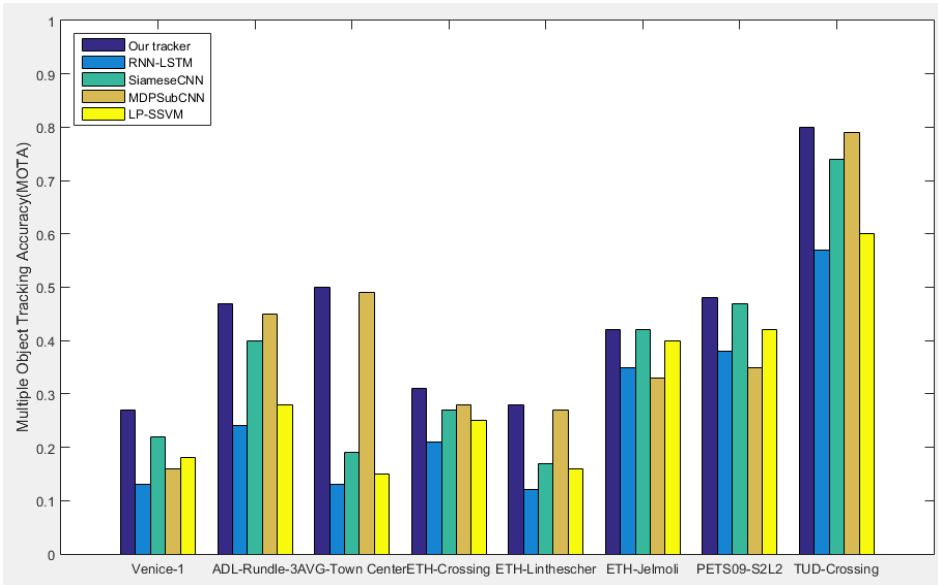


FIGURE 4. The comparison results of MOTA on the MOT Challenge dataset

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901300, IEEE Access

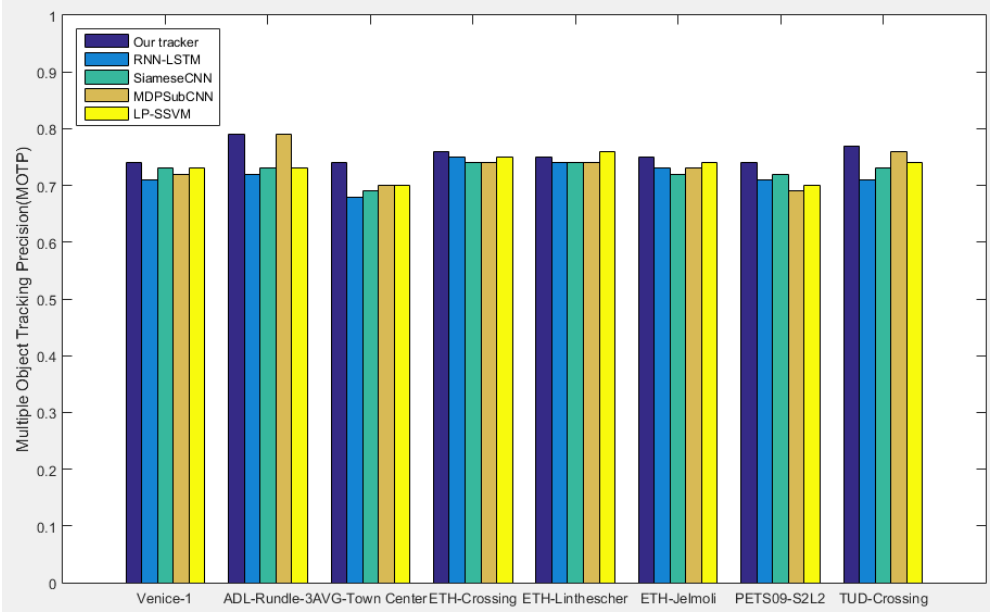
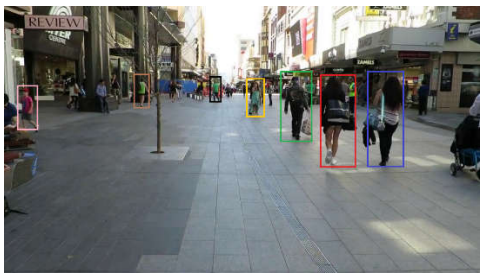


FIGURE 5. **The comparison results of MOTP on the MOT Challenge dataset**

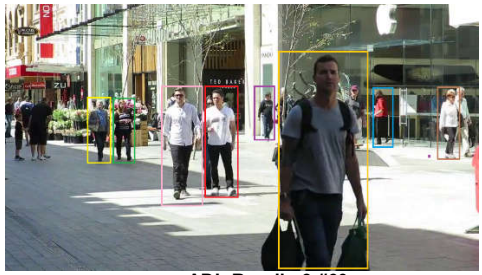
C. Qualitative Evaluation

Due to the limited given space, we only list the part of tracking results on the test videos in the MOT challenge benchmark, as demonstrated in Figure 6.

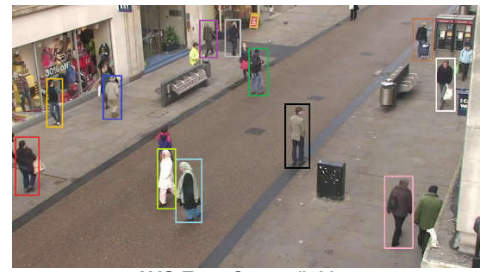
Venice-1 #60



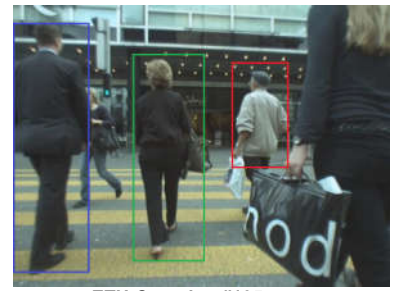
ADL-Rundle-1 #240



ADL-Rundle-3 #60



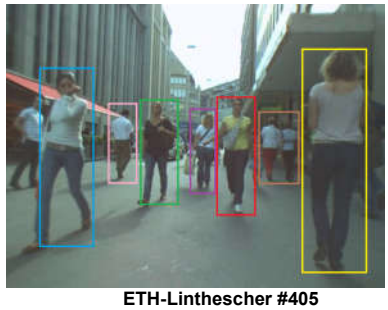
AVG-TownCentre #420



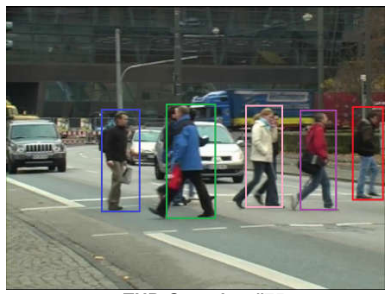
ETH-Crossing #105

ETH-Jelmoli #255

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901300, IEEE Access



ETH-Linthescher #405



TUD-Crossing #75

PETS09-S2L2 #240

FIGURE 6. Sample tracking results on the MOT Challenge benchmark

The above experimental data listed in the Figure 4-6 demonstrate the superior performance of our track strategy with MADRL in both precision and success rate.

V. CONCLUSION

There are some problems in the existing multi-object trackers, for example, they fails when the object emerging or disappearing, there are many limitations as complex characteristics of the objects can not be captured by the hand-crafted features, the tracked objects have similar appearance , etc.. To overcome these problems, a novel multi-object tracking approach based on MADRL was proposed in this paper. The object detector YOLO V3 was adopted to detect the multiple objects. The detected results is considered as multiple agents, then, we adopt a MADRL to obtain a joint action for multiple objects and get the multi-object tracking results. The experimental results showed that the proposed multi-object tracking method obtains the better performances in the robustness and accuracy.

REFERENCES

[1]. Son J, Baek M, Cho M, et al. Multi-object Tracking with Quadruplet Convolutional Neural Networks[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2017:3786-3795.

[2]. Jiang, M. X.; Pan, Z. G.; Tang, Z. Z. Visual Object Tracking Based on Cross-Modality Gaussian-Bernoulli Deep Boltzmann Machines with RGB-D Sensors. *Sensors*, 2017, 17,121-138.

[3]. Ak, K. C.; Jacques, L.; De, V. C. Discriminative and efficient label propagation on complementary graphs for multi-object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* 2017, 39, 61-74.

[4]. Rosario J R B D, Bandala A A, Dadios E P. Multi-view multi-object tracking in an intelligent transportation system: A literature review[C]// IEEE, International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management. IEEE, 2017:1-4.

[5]. Naiel, M. A.; Ahmad, M. O.; Swamy, M. N. S., et al. Online multi-object tracking via robust collaborative model and sample selection. *Computer Vision & Image Understanding*, 2017, 154:94-107.

[6]. Shitrit H B, Berclaz J, Fleuret F, et al. Multi-Commodity Network Flow for Tracking Multiple People[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2013, 36(8):1614-27.

[7]. Schulter S, Vernaza P, Choi W, et al. Deep Network Flow for Multi-object Tracking[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2017:2730-2739.

[8]. Ren, S.; He, K.; Girshick, R., et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017, 39: 1137 - 1149.

[9]. Andriyenko, A.; Schindler, K. Multi-target tracking by continuous energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016, 38:2054-2066.

[10]. Girshick R. Fast R-CNN. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015:1440-1448.

[11]. Redmon, J., Divvala, S., Girshick, R., Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016:779-788.

[12]. Milan A, Roth S, Schindler K. Continuous energy minimization for multitarget tracking[J]. IEEE Trans Pattern Anal Mach Intell, 2014, 36(1):58-72.

[13]. Henriques, J. F.; Caseiro, R.; Batista, J. Globally optimal solution to multi-object tracking with merged measurements. In Proceedings of the IEEE Conference on Computer Vision (ICCV). 2011:2470-2477.

[14]. Butt, A. A.; Collins, R. T. Multi-target Tracking by Lagrangian Relaxation to Min-cost Network Flow. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013:1846-1853.

[15]. Thangali A. Coupling detection and data association for multiple object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012:1948-1955.

[16]. Bae, S. H.; Yoon, K. J. Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking. IEEE Trans. Pattern Anal. Mach. Intell. 2017.

[17]. Wen, L.; Lei, Z.; Lyu, S., et al. Exploiting Hierarchical Dense Structures on Hypergraphs for Multi-Object Tracking. IEEE Trans. Pattern Anal. Mach. Intell. 2016, 38(10):1983-1996.

[18]. Breitenstein, M. D.; Reichlin, F.; Leibe, B. et al. Online Multiperson Tracking-by-Detection from a Single, Uncalibrated Camera. IEEE Trans. Pattern Anal. Mach. Intell. 2011, 33(9):1820.

[19]. He, Z.; Li, X.; You, X., et al. Connected Component Model for Multi-Object Tracking. IEEE Trans. Image Processing, 2016, 25(8):3698-3711.
[20]. Dehghan, A.; Tian, Y.; Torr, P. H. S., et al. Target Identity-aware Network Flow for online multiple target tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015:1146-1154.
[21]. Viola, P.; Jones, M.J. Robust real-time face detection. Int. J. Comput. Vis.2004, 57, 137–154.
[22]. Navneet, D.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 20–26 June 2005.
[23]. Wang, X.Y.; Han, T.X.; Yan, S.C. An HOG-LBP human detector with partial occlusion handling. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Kyoto, Japan, 29 September–2 October 2009.
[24]. A. Kai, M. P. Deisenroth, M. Brundage, et al. "Deep Reinforcement Learning: A Brief Survey". IEEE Signal Processing Magazine, vol.34, no.6,pp,26-38, 2017.
[25]. R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction. Cambridge, MA: MIT Press, 1998.
[26]. N. Kohl and P. Stone, "Policy gradient reinforcement learning for fast quadrupedal locomotion," in Proc. IEEE Int. Conf. Robotics and Automation, 2004, pp. 2619–2624.
[27]. Y. Li. (2017). Deep reinforcement learning: An overview. arXiv. [Online].Available: https://arxiv.org/abs/1701.07274.
[28]. Jayaraman, D.; Grauman, K. Look-ahead before you leap: end-to-end active recognition by forecasting the effect of motion. arXiv:1605.00164, 2016.
[29]. Silver, D. ; Huang, A.; Maddison, C. J. ; Guez, A.; Sifre, L. ; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V. ; Lanctot, M. et al. Mastering the game of go with deep neural networks and tree search[J]. Nature, 529(7587):484–489, 2016.
[30]. Zhang, D.; Maei, H.; Wang, X., et al. Deep Reinforcement Learning for Visual Object Tracking in Videos. arXiv preprint, 2017.
[31]. Luo, W.; Sun, P.; Mu, Y., et al. End-to-end Active Object Tracking via Reinforcement Learning. arXiv preprint, 2017.
[32]. Yun, S.; Choi, J.; Yoo, Y.; Yun, K.; Choi, J. Y. Action-Decision Networks for Visual Tracking with Deep Reinforcement Learning, CVPR 2017.
[33]. Jayaraman, D.; Grauman, K. Look-ahead before you leap: end-to-end active recognition by forecasting the effect of motion. arXiv:1605.00164, 2016.
[34]. Jie, Z.; Liang, X.; Feng, J. et al. Tree-Structured Reinforcement Learning for Sequential Object Localization. In Advances in Neural Information Processing, 2016:127-135.
[35]. Caicedo, J. C.; Lazebnik. S. Active object localization with deep reinforcement learning. CVPR, 2015: 2488–2496.
[36]. Bloembergen D, Tuyls K, Hennes D, et al. Evolutionary dynamics of multi-agent learning: a survey[J]. Journal of Artificial Intelligence Research, 2015, 53(1):659-697.
[37]. Buşoniu L, Babuška R, Schutter B D. Multi-agent Reinforcement Learning: An Overview[J]. Studies in Computational Intelligence, 2010, 310:183-221.
[38]. Noureddine D B, Gharbi A, Ahmed S B. Multi-agent Deep Reinforcement Learning for Task Allocation in Dynamic Environment[C]// International Conference on Software Technologies. 2017:17-26.
[39]. R. Bellman, "On the theory of dynamic programming," Proc. Nat. Acad. Sci., vol.38, no. 8, pp. 716–719, 1952.
[40]. Joseph Redmon, Ali Farhadi, YOLOv3 : An Incremental Improvement, https://pjreddie.com/media/files/papers/YOLOv3.pdf.
[41]. X. Kong, B. Xin, Y. Wang and G. Hua, "Collaborative Deep Reinforcement Learning for Joint Object Search," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, USA, 2017, pp. 7072-7081.
[42]. Xiang, Y.; Alahi, A.; Savarese, S. Learning to Track: Online Multi-Object Tracking by Decision Making. In International Conference on Computer Vision (ICCV), pp. 4705-4713, 2015.
[43]. Milan, A.; Rezatofighi, S. H.; Dick, A. et al. Online Multi-Target Tracking Using Recurrent Neural Networks. AAAI 2017.
[44]. Leal-Taixé, L.; Canton-Ferrer, C.; Schindler, K. Learning by Tracking: Siamese CNN for Robust Target Association. DeepVision Workshop (CVPR), Las Vegas (Nevada, USA), June 2016.
[45]. Wang, S.; Fowlkes, C. Learning Optimal Parameters for Multi-target Tracking with Contextual Interactions. In International Journal of Computer Vision, 2017,122(3):484—501.
[46]. Jiang, M. X.; Deng Ch.; Pan, Z. G.; Chen X.; Wang L. F.; and Sun X.. Multiple Object Tracking in Videos Based on LSTM and Deep Reinforcement Learning [J],Complexity, 2018, Article ID 4695890(Online).
[47]. Leal-Taixé, L.; Milan, A.; Reid, I.; Roth, S. & Schindler, K. MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. arXiv:1504.01942.
[48]. Rezatofighi S H, Milan A, Zhang Z, et al. Joint Probabilistic Data Association Revisited[C]// IEEE International Conference on Computer Vision. IEEE, 2015:3047-3055.

MINGXIN JIANG received her B.S. degree in Measurement & Control Technology and Instrument and the M.S. degree in Communications and Information System from Jilin University, Changchun, China, in 2002 and 2005. She received a Ph.D. degree in Signal and information processing, Dalian University of Technology, China, in 2013. She was a post-doctoral researcher with the Department of Electrical Engineering in Dalian University of Technology from 2013 to 2015. She is currently an associate professor in Faculty of Electronic information Engineering at Huaiyin Institute of Technology. Her research interests include multi-object tracking, video content analysis and vision sensors for robotics.

CHAO DENG received the B.S. degree and the M.S. degree in communication engineering from Jilin University, China, in 2002 and 2005.He then received the Ph.D. degrees in Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences in 2008.He is currently an associate professor with the School of Physics and Electronic Information Engineering, Henan Polytechnic University, Jiaozuo China. His research interests include image processing and signal processing.

YINSHAN YU received PhD from Nanjing University of Aeronautics and Astronautics (NUAA), China, in 2017. Since 2017, he has been a lecturer in Huaiyin Institute of Technology (HYIT), China. His research interests include RFID dynamical test and image processing.

JINGSONG SHAN is a lecturer at Huaiyin Institute of Technology, China. He received his M.S. degree from Guizhou University in 2006, and his Ph.D. degrees PLA University of Science and Technology in 2017 . His current research interests include information retrieval, random algorithm and machine learning.