Received October 30, 2020, accepted November 10, 2020, date of publication December 21, 2020, date of current version January 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3046243

A Blockchain-Enabled Multiple Object Tracking for Unmanned System With Deep Hash Appearance Feature

ZHENGUO DING[1], SITONG LIU[1], MIN LI[1,2,3], ZHICHAO LIAN[1], (Member, IEEE), AND HUI XU[4]

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
[2]Key Laboratory of Biorheological Science and Technology, Ministry of Education, Chongqing University, Chongqing 400044, China
[3]Shandong Key Laboratory of Digital Medicine and Computer Assisted Surger, The Affiliated Hospital of Qingdao University, Qingdao 266003, China
[4]Samsung Electronics (China) Research and Development Center, Nanjing 210018, China

Corresponding author: Zhichao Lian (lzcts@163.com)

This work was supported in part by the 4th project ''Research on the Key Technology of Endogenous Security Switches'' under Grant 2020YFB1804604; in part by the National Key Research and Development Program ''New Network Equipment Based on Independent Programmable Chips'' under Grant 2020YFB1804600; in part by the 2020 Industrial Internet Innovation and Development Project from Ministry of Industry and Information Technology of China; in part by the Fundamental Research Fund for the Central Universities under Grant 30918012204, Grant 30919011231, and Grant 30919011401; in part by the Science and Technology project in Jiangsu Province under Grant BE2016904 and Grant BE2017739; in part by the National Natural Science Foundation of China under Grant 61501241; in part by the Visiting Scholar Foundation of Key Laboratory of Biorheological Science and Technology (Chongqing University), Ministry of Education under Grant CQKLBST-2018-011; and in part by the Foundation of Shandong provincial Key Laboratory of Digital Medicine and Computer assisted Surgery under Grant SDKL-DMCAS-2018-04.

ABSTRACT The unmanned system based on mobile-edge computing (MEC) and blockchain can conveniently share computing resources and realize multi-devices collaboration. However, frequent data communication in the framework of object detection brings a heavy computational burden. In this paper, a novel blockchain-based multi-view unmanned equipment fusion architecture using multiple object tracking (MOT) technique is designed. Inspired by the idea of person re-identification and hash representation for image retrieval, a novel MOT technique using HashNet to extract deep hash appearance of objects is proposed. In addition, based on the blockchain and MEC technology, we make some improvements in feature fusion and tracking interrupt recovery. We combine deep hash appearance features with motion features and design a tracking interruption recovery mechanism to solve the problem of object occlusion. The experiment results on the MOT challenge dataset demonstrate that the proposed algorithm can handle object occlusion problem effectively and successfully reduce the number of identity switches. In real application scenes our algorithm performs particularly well, showing that our algorithm is more practical.

INDEX TERMS Multiple object tracking, computer vision, blockchain, hash representation, person re-identification, machine learning.

I. INTRODUCTION

Distributed Deep Learning (DDL) has been one of the most attractive research areas now. The collaboration of multiple unmanned devices based on the DDL helps us solve the problem of multi-view object detection and tracking, and it ensures the exchange of information at the different time and space. The key problem for unmanned system using the DDL is to balance computational burden and efficiency. Jiang *et al.* [36] proposed a novel framework of applying mobile-edge computing (MEC) and blockchain to implement domain-adaptive object detection for multiple devices. In this framework, firstly the MEC is a technology used in mobile communication systems, edge nodes, and undertaking a large number of computing tasks. Second, blockchain is a new application mode of computer technology such as distributed data storage, point-to-point transmission, consensus mechanism, and encryption algorithm. However, object detection in a multi-device environment proposed in this framework will bring frequent data communication, and will cause a heavy computational burden on the entire system

Object detection cannot fully meet the requirements of unmanned systems because of the heavy burden on communication and computation. Therefore, a novel blockchain-based

The associate editor coordinating the review of this manuscript and approving it for publication was Zhipeng Cai.

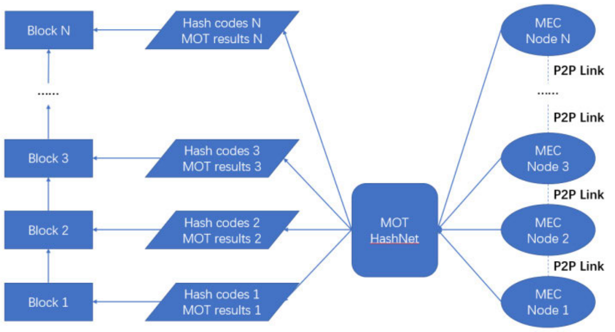

FIGURE 1. Example of system architecture.

FIGURE 2. Example of direction change.

FIGURE 3. Example of target occlusion.

multi-view unmanned equipment collaborative fusion architecture is proposed. In this architecture, multiple object tracking techniques for unmanned system is applied based on blockchain [20]–[24], [30]–[35] and mobile-edge computing (MEC) technologies [17]–[19]. It is essentially a database, which has the characteristics of decentralization, non-tampering, full trace retention, traceability, collective maintenance, openness and transparency, etc., which can solve the problem of information asymmetry and achieve collaborative trust and consistency between multiple subjects action. Moreover, the existing multiple object tracking (MOT) algorithms are not well adapted to the blockchain system, so we designed a new MOT algorithm that can better use the data, models and computing resources managed by the decentralized coordination platform of the blockchain.

The MOT problem can be considered as a multivariate estimation problem, whose purpose is to associate objects of adjacent frames. So how to associate objects accurately is one of the key problems in the MOT. During the development of MOT algorithms, lots of features have been used in data association module to measure similarity of objects. In early period, simple object were widely used, such as motion feature Intersection-over-Union (IOU), appearance feature Histogram of Oriented Gradient (HOG) [2] and so on. However, motion feature can only perform well in simple scenes where objects move regularly. The traditional appearance feature cannot make a perfect distinction among objects. In recent years, with the development of deep learning, more and more MOT algorithms use AlexNet [3], RNN [4], ResNet [5], VGG16 [6] and other deep neural networks to extract deep feature of objects to improve the accuracy of data association.

For higher tracking speed, some MOT algorithms such as [7] and SORT [8] did not use appearance information of object. These algorithms abandoned the detail information, and only utilized position information of objects for data association. However, in many scenarios, the movement of objects is irregular and non-linear, which causes obvious deviations in the position information of objects. In a word, the MOT algorithm without appearance information of objects cannot perform well in complex scenes and it cannot handle object occlusion problem as well.

With the development of deep learning, deep features can effectively differentiate objects. Using deep appearance features in MOT algorithms has become a trend. DEEP SORT [9] employed a wide residual network with two convolutional layers followed by six residual blocks to extract features of objects, and this method reduced the number of identity switches successfully. POI [10] applied a superior detector and utilizes a neural network which is similar to GoogleNet [11] to extract appearance features, as a result it achieved a better performance than the state-of-the-art algorithms.

In order to get a better performance, in this paper we combine motion feature with deep hash appearance feature to solve object occlusion and interaction problem. Inspired by the idea of person re-identification and hash representation for image retrieval [25]–[29], we connect the classical person re-identification network ResNet with a fully connected layer of hash to construct a deep hash network HashNet, and then we use HashNet to extract deep hash appearance features of human objects. Compared with the traditional deep neural networks, HashNet can better differentiate objects of the same category which can improve the accuracy of MOT. In addition, we also design an interruption recovery mechanism to re-track the failure trajectory. Finally, the experiment result shows that our algorithm can handle object occlusion and tracking interruption effectively.

II. FEATURE REPRESENTATION

This section introduces the feature representation method of our algorithm.

A. APPEARANCE FEATURE

In realistic scenarios, objects occlusion and interaction is common. In order to have a better performance against tracking interruption, we integrate object appearance features with motion features in our algorithm. Multiple object tracking mainly focuses on multi-person tracking problem, whose purpose is to associate persons of the same identity. However, no matter the identities of persons are same or not, they both belong to the same kind of objects. Therefore, the traditional deep features used for classification problem are not applicable in this situation.

Association among persons of different identities is similar to people re-identification problem. In this paper, we use ResNet50 as the base network to extract appearance feature of people. The base network is pre-trained on ImageNet [12] and its input size is 224∗224. In addition, we slightly adjust the last three layers of ResNet50 and then fine-tuning the whole network on Market-1501[13]. During the training phase we utilize 12,936 images of 751 identities and we finally obtained Rank@1 accuracy 88.84% and mAP 71.59% on the Market-1501 test dataset.

Hash codes are widely applied in image retrieval [14] and learning to hash is widely applied in approximate nearest neighbor (ANN) [15]. The purpose of ANN is to find the most similar top N targets from a massive database which is similar to data association in multiple object tracking. Therefore, we apply the thought of hash in our tracking framework to further improve the accuracy of data association. We combine ResNet50 with a fully connected layer of hash to construct HashNet, whose flow chart is shown in Figure 4.

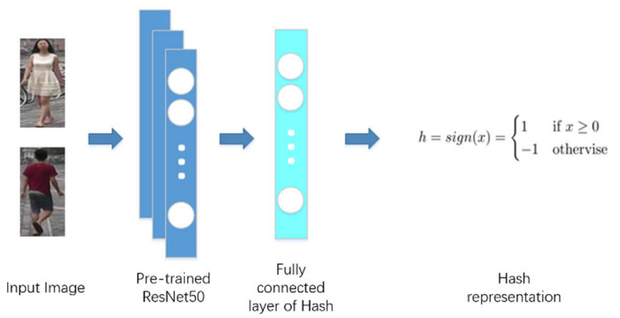

FIGURE 4. The flow chart of HashNet.

When an image vector is put into HashNet, firstly we extract its features by the pre-trained ResNet50 which is trained on Market-1501. After that the features will be transmitted into the fully-connected layer of hash so that the feature can be converted to a K-dimensional representation. In the end, we convert the K-dimensional representation of feature to a K-bit hash encoding by activation function $h = sign(x)$.

However, if the input of $h = sign(x)$ is non-zero, the gradient of the function is zero, which would lead to gradient disappearance. Therefore we apply the smooth function $h = tanh(x)$ instead of the original non-smooth function $h = sign(x)$ to make it more easier to optimize. As formula (1) shows, in order to solve the problem of gradient disappearance, during training process we gradually reducing the smoothness of $h = tanh(x)$ by increasing ξ to make $h = tanh(x)$ gradually approach the original function $h = sign(x)$.

$$\lim_{\xi \to \infty} tanh\,(\xi x) = sign\,(x) \tag{1}$$

The HashNet is trained on Mars [16] train dataset which contains 625 identities of people, and totally 509914 images and we set hash code bits $T = 80$. In addition, we use inner product to measure the similarity between two hash codes as formula (2). The larger the inner product is, the higher the similarity is.

$$\begin{aligned} hashcode_x &= \{x_1, x_2, \ldots, x_T\} \\ hashcode_y &= \{y_1, y_2, \ldots, y_T\} \\ InnSim\,(x, y) &= \sum_{i=1}^{T} < x_i, y_i > \div T \end{aligned} \tag{2}$$

We select several pictures from Mars, including pedestrians with the same ID and different IDs as shown in Figure 5 to demonstrate the performance of HashNet. We calculate the feature distance of ResNet and HashNet respectively, where feature distance equals to 1 minus feature similarity. From Table 1 we can see that HashNet has a lager gap between feature distances of different persons than ResNet50, which means HashNet can make a better distinction of people with different identities.

FIGURE 5. Mars dataset validation examples.

TABLE 1. Feature distance comparison results between ResNet50 and HashNet.

Person ID	Network	a	b	c	d	e	f	g
1	ResNet50	0.041	0.083	0.062	0.081	0.047	0.053	0.071
1	HashNet	0.225	0.250	0.375	0.475	0.300	0.350	0.425
2	ResNet50	0.147	0.153	0.162	0.134	0.144	0.213	0.282
2	HashNet	0.600	0.575	0.800	0.675	0.925	0.675	0.425
3	ResNet50	0.253	0.211	0.171	0.376	0.371	0.392	0.413
3	HashNet	0.775	0.725	0.750	0.450	0.600	0.725	0.600

B. FEATURE FUSION

Motion features are necessary in MOT algorithms. It can perfectly use object position information which is important in data association. At the same time, appearance features are more robust and it can deal with object occlusion and tracking interruption effectively. So in this paper we integrate motion with appearance features by setting appropriate weights to achieve a more robust feature representation.

In this paper we take the Kalman filter to predict location of each object in current frame. We utilize IOU as the motion feature of object to measure relevance of different objects. The motion feature similarity between objects is calculated as follow:

$$Sim_{mot}\left(trk_i, det_j\right) = \frac{Area(trk_i) \cap Area(det_j)}{Area(trk_i) \cup Area(det_j)} \tag{3}$$

where $det = \{det_j\}_j^M$ represents the bounding boxes set of detected objects in current frame, det_j means the $j-th$ detection bounding boxes of M detections, $trk = \{trk_i\}_i^N$ represents the predicted positions set of objects in next frame, trk_i means the $i-th$ predicted bounding boxes of N trajectories.

To associate objects which have been occluded for few frames to the original trajectory rather than treating it as a new object, we should use multi-frame appearance features instead of only use the features of adjacent frame. In this paper we use a feature pool Fea_{id}^K to save appearance features of all trajectories for a few frames, where K means the capacity of feature pool, and id means the id of trajectory. The feature pool would preserve new features in time. When the feature pool is full, new features will replace the original features which exist in the pool with the longest storage time. Comparing to setting multiple feature vectors for each trajectories, the feature pool can ensure tracking speed and avoid complex calculation when the number of objects in current frame is large. When calculate inner product distance between det_j and trk_i, we select the feature of trajectories in Fea_{id}^K which $id = i$ as Fea_{trk_i}, and then choose the maximum similarity between det_j and Fea_{trk_i} as the final appearance similarity between det_j and trk_i:

$$\begin{aligned} Sim_{app}\left(trk_i, det_j\right) = MAX(&InnSim(det_j, Fea_{trk_i}| \\ &Fea_{trk_i} \in Fea_{id=i}^K)) \end{aligned} \tag{4}$$

For feature fusion, both motion feature and appearance feature play important roles in data association of MOT. Firstly, motion features applied the position information of the objects which is particular useful when motive uncertain is low. On the other hand, robust appearance features of objects can effectively solve object occlusion problem and it is helpful in re-tracking objects of failure trajectories. So in this paper we integrate motion and appearance features by weight as formula (5). Through this way we obtain a robust feature representation of objects which is adapt to complex scenes:

$$\begin{aligned} Sim\left(trk_i, det_j\right) &= \alpha \times Sim_{mot}\left(trk_i, det_j\right) \\ &\quad + (1-\alpha) \times Sim_{app}\left(trk_i, det_j\right) \end{aligned} \tag{5}$$

$Sim\left(trk_i, det_j\right)$ is employed as cost matrix of Hungarian Algorithm in association module, and the parameter α can be adjusted to be suitable for different scenarios. If there is no frequent integration between objects and the movement of object is regular, α can be increased. On the contrary, α will be decreased in complex scenarios with massive occlusion.

III. TRACKING ALGORITHM

In this section, we mainly introduce the overall flow of the algorithm and parameters setting. The overall algorithm flow is described as Algorithm 1.

Algorithm 1 Multiple Object Tracking With Deep Hash Appearance Feature

Input:
The $t-th$ frame of video sequence. The detection objects set of frame t: D^t. The trajectory set of frame $t-1$: T^{t-1}.
Output:
New trajectory set of frame t: T^t.
1. Calculate the $Sim(det_j, trk_i)$ of det_j in D^t and predict bounding box trk_i of T^{t-1}, save as cost matrix C.
2. Use Hungarian algorithm to make optimal match between D^t and T^{t-1} based on cost matrix C.
3. Check match result, if IOU of det_j and trk_i is less than IOU threshold β and appearance similarity of them is less than appearance threshold γ, consider they don't match;
4. Recording the remaining time of unmatched trk_i, if the time is longer than trace life threshold θ, consider that the object of this trajectory has disappeared, the track of this trajectory is over.
5. After data association, update the trajectory set T^{t-1} with matched det in $t-th$ frame, save the new trajectory set as T^t.

Firstly, it is known from experience that the positions of objects could not change a lot in adjacent frames, so if IOU of trk_i and det_j is less than a threshold β we consider they don't match. Secondly, in order to match the object that reappears after a long period of occlusion to the original trajectory, we need to keep unmatched trajectory for several frames, and utilize a threshold θ to control the lifetime of unmatched trajectory. If the trajectory has not been successfully matched during the lifetime we consider that the tracking of this trajectory is completely over. Thirdly, when object reappears after a long-term occlusion, the current position of object would be far away from the last appearing position. In this situation, although we re-track the object, it will be deleted from the original trajectory due to the limit of IOU. Therefor we apply a cascade judgement method. When IOU of trk_i and det_j is

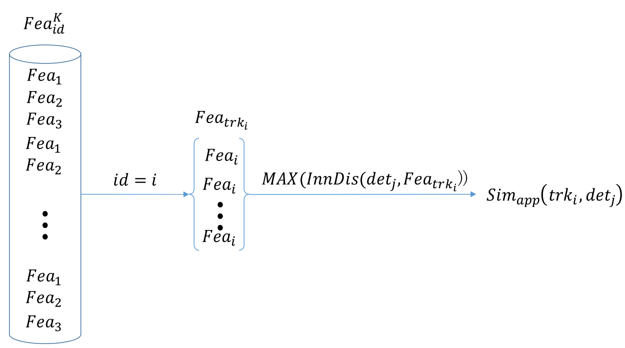

FIGURE 6. Feature calculation flowchart.

FIGURE 7. MOT15 data set example.

less than β, we keep calculating the appearance similarity of them. If the similarity is higher than a threshold γ, we determine det_j still matched with trk_i.

IV. EXPERIMENT

We evaluate the performance of our algorithm on MOT challenge datasets [16]. MOT challenge is a dataset for evaluating the performance of MOT algorithms. The MOT15 data set contains 22 video clips, 11 of which are training sets and 11 are test sets. Each video clip includes a video image sequence, a target detection position annotation file, and a target true position annotation file. On the other hand, as shown in Figure 6, the data set includes a variety of camera movement methods, different shooting angles, and diverse weather to simulate richer application scenarios. The experiments in POI [10] prove that object detection results have a significant impact on the performance of MOT algorithm. To avoid the influence of detector, in the comparison experiments we utilize the same detector for all algorithms. The experiment results are shown in Table 2.

TABLE 2. Tracking results on MOT15 train datasets.

Algorithm	MT↑	ML↓	FP↓	FN↓	ID Sw.↓	MOTA↑	MOTP↑
KCF_IOU	22.8%	29.6%	5732	17631	577	38.3	72.7
SORT [8]	24.0%	29.0%	5871	17573	544	39.9	72.8
SORT_CNN [9]	24.2%	27.6%	6087	17731	458	39.7	72.8
Ours_VGG16	23.6%	28.8%	6224	17848	498	39.6	72.7
Ours_ResNet50	23.6%	28.4%	6041	17831	473	39.0	72.6
Ours_HashNet	24.2%	26.8%	6093	17676	433	40.5	72.8

To ensure that the experimental results are true and effective, this article adopts the official MOT evaluation method, and the evaluation indicators are described as follows:

(1) MT Mostly tracked targets. The ratio of ground-truth trajectories that are covered by a track hypothesis for at least 80% of their respective life span.

(2) ML Mostly lost targets. The ratio of ground-truth trajectories that are covered by a track hypothesis for at most 20% of their respective life span.

(3) FP The total number of false positives.

(4) FN The total number of false negatives (missed targets).

(5) ID Sw. Number of Identity Switches (ID switch ratio = #ID switches / recall) [3]. Please note that we follow the stricter definition of identity switches as described in the reference.

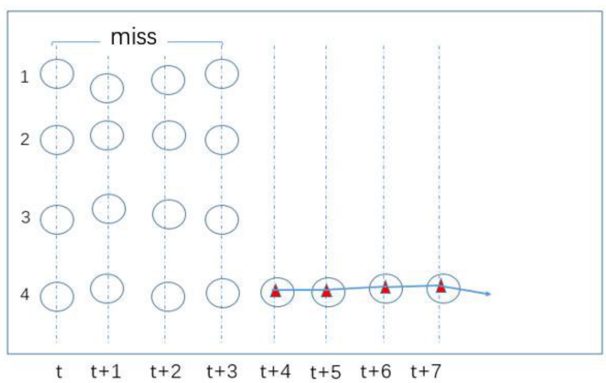

FIGURE 8. Tracking missed examples.

TABLE 3. Tracking results on PETS09-S2L1 video sequence.

Algorithm	MT↑	ML↓	FP↓	FN↓	ID Sw.↓	MOTA↑	MOTP↑
SORT	42.1%	0	485	1119	102	61.9	67.7
Ours_HashNet	47.4%	0	470	1075	43	64.5	67.9

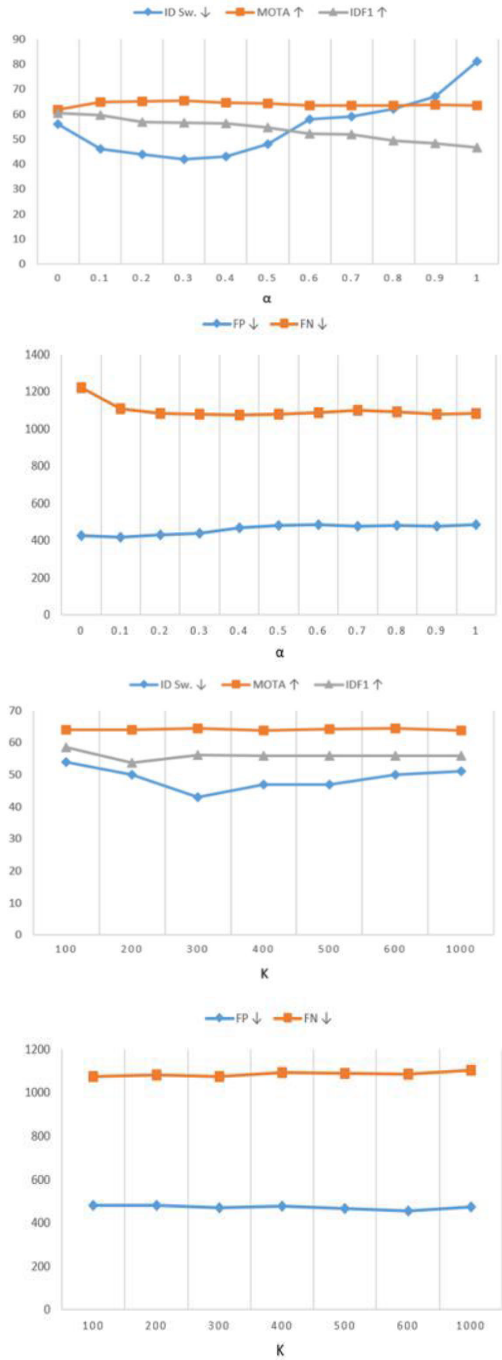

FIGURE 9. Result of parameter comparison experiment.

(6) MOTA Multi-Object Tracking Accuracy (+/- denotes standard deviation across all sequences). This measure combines three error sources: false positives, missed targets and

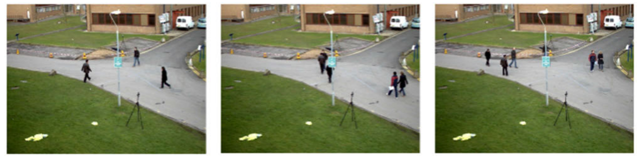

FIGURE 10. PETS09-S2L1 video sequence example.

identity switches.

$$MOTA = 1 - \frac{\sum_t (FP + FN + IDSw.)}{\sum_t gt} \tag{6}$$

(7) MOTP Multi-Object Tracking Precision (+/- denotes standard deviation across all sequences). The misalignment between the annotated and the predicted bounding boxes.

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \tag{7}$$

It is worth noting that when calculating MOTA and MOTP, the average value should be calculated in the cycle of tracking the entire process, rather than calculating the result of each frame. If the value is calculated by frame and finally averaged instead of the tracking period, then the result will be inconsistent with expectations.

From the experiment results we can see, the proposed algorithm successfully reduced the number of identity switches. Comparing with the classic algorithm SORT, the number of ID Sw. has been reduced from 544 to 433. The consequence certificates our method can effectively handle problem of object occlusion and interaction, but at the same time the frequent processing of object occlusion also brings an increase in the number of track fragmentation (FM). In addition, we found that our method performs particular well in fixed camera video sequence. PETS09-S2L1 is a video sequence of MOT15 which is caught by a fixed camera at high altitude. It is a well-known dataset, targeted primarily at surveillance applications. The results on PETS09-S2L1 video sequence in Table 3 show that the number of ID Sw. has been reduced from 102 to 43, decreased by 57.8%, and other indicators also have a significant increase. That's mainly because when camera is moving, the scale of object bounding boxes will change frequently which will cause huge errors in deep hash appearance features. In real application scenarios, positions of cameras are generally fixed which is similar to PETS09-S2L1, so the excellent result on PETS09-S2L1 also indicate that out algorithm is more practical. Apart from this, we evaluated the effects of parameter α and K on the experimental results. The results are shown as Figure 9, it can be seen that parameter α and K do not affect MOTA obviously but they have huge effects on ID Sw.

V. CONCLUSION

In this paper, we propose a new blockchain-based multi-view collaborative fusion architecture for unmanned system using MOT techniques. To improve the performance of MOT algorithm. we construct a HashNet to extract deep hash appearance features of pedestrians which has a better discrimination than traditional deep features. With feature fusion and tracking interruption recovery mechanism we successfully handle tracking interruption caused by object occlusion and interaction, and reduce the number of identity switches obviously. In future, we will continue to simplify the network structure of HashNet and improve the feature fusion method to obtain more accurate tracking results.

REFERENCES

[1] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, X. Zhao, and T. Kim, "Multiple object tracking: A literature review," 2014, *arXiv:1409.7618*. [Online]. Available: https://arxiv.org/abs/1409.7618

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2005, pp. 886–893.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[4] A. Milan, S. Hamid Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," 2016, *arXiv:1604.03635*. [Online]. Available: http://arxiv.org/abs/1604.03635

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

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

[7] E. Bochinski, V. Eiselein, and T. Sikora, "High-speed tracking-by-detection without using image information," in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2017, pp. 1–6.

[8] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3464–3468.

[9] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3645–3649.

[10] F. Yu, W. Li, Q. Li, Yu Liu, X. Shi, and J. Yan, "POI: Multiple object tracking with high performance detection and appearance feature," in *Proc. ECCV Workshops*, 2016, pp. 36–42.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[13] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1116–1124.

[14] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Proc. CVPR*, Jun. 2011, pp. 817–824.

[15] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in *Proc. 47th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, Oct. 2006, pp. 459–468.

[16] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," 2016, *arXiv:1603.00831*. [Online]. Available: http://arxiv.org/abs/1603.00831

[17] Y. He, F. Yu, N. Zhao, and H. Yin, "Secure social networks in 5G systems with mobile edge computing, caching, and device-to-device communications," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 103–109, Jun. 2018.

[18] M. Liu, F. Yu, Y. Teng, V. C. Leung, and M. Song, "Computation offloading and content caching in wireless blockchain networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 11008–11021, Nov. 2018.

[19] C-M. Wang, F. Yu, C. Liang, Q. Chen, and L. Tang, "Joint computation offloading and interference management in wireless cellular networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 7432–7445, Aug. 2017.

[20] X. Chi, C. Yan, H. Wang, W. Rafique, and L. Qi, "Amplified locality-sensitive hashing-based recommender systems with privacy protection," *Concurrency Comput., Pract. Exper.*, p. 5681, 2020.

[21] W. Zhong, X. Yin, X. Zhang, S. Li, W. Dou, R. Wang, and L. Qi, "Multi-dimensional quality-driven service recommendation with privacy-preservation in mobile edge environment," *Comput. Commun.*, vol. 157, pp. 116–123, May 2020.

[22] L. Qi, C. Hu, X. Zhang, M. R. Khosravi, S. Sharma, S. Pang, and T. Wang, "Privacy-aware data fusion and prediction with spatial-temporal context for smart city industrial environment," *IEEE Trans. Ind. Informat.*, early access, Jul. 28, 2020, doi: 10.1109/TII.2020.3012157.

[23] L. Qi, X. Wang, X. Xu, W. Dou, and S. Li, "Privacy-aware cross-platform service recommendation based on enhanced locality-sensitive hashing," *IEEE Trans. Netw. Sci. Eng.*, early access, Jan. 27, 2020, doi: 10.1109/TNSE.2020.2969489.

[24] L. Wang, X. Zhang, R. Wang, C. Yan, H. Kou, and L. Qi, "Diversified service recommendation with high accuracy and efficiency," *Knowl.-Based Syst.*, vol. 204, Sep. 2020, Art. no. 106196.

[25] W. Lu, Y. Li, Y. Cheng, D. Meng, B. Liang, and P. Zhou, "Early fault detection approach with deep architectures," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 7, pp. 1679–1689, Jul. 2018.

[26] X. Zhou, W. Liang, K. Wang, H. Wang, L. T. Yang, and Q. Jin, "Deep-learning-enhanced human activity recognition for Internet of healthcare things," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6429–6438, Jul. 2020.

[27] X. Zhou, Y. Li, and W. Liang, "CNN-RNN based intelligent recommendation for online medical pre-diagnosis support," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, early access, May 14, 2020, doi: 10.1109/TCBB.2020.2994780.

[28] X. Zhou, W. Liang, K. Wang, R. Huang, and Q. Jin, "Academic influence aware and multidimensional network analysis for research collaboration navigation based on scholarly big data," *IEEE Trans. Emerg. Topics Comput.* early access, Jul. 26, 2018, doi: 10.1109/TETC.2018.2860051.

[29] X. Zhou, W. Liang, K. Wang, and S. Shimizu, "Multi-modality behavioral influence analysis for personalized recommendations in health social media environment," *IEEE Trans. Comput. Social Syst.*, vol. 6, no. 5, pp. 888–897, Oct. 2019.

[30] S. Zhu, Z. Cai, H. Hu, Y. Li, and W. Li, "zkCrowd: A hybrid blockchain-based crowdsourcing platform," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4196–4205, Jun. 2020.

[31] S. Zhu, W. Li, H. Li, L. Tian, G. Luo, and Z. Cai, "Coin hopping attack in blockchain-based IoT," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4614–4626, Jun. 2019.

[32] Y. Liang, Z. Cai, J. Yu, Q. Han, and Y. Li, "Deep learning based inference of private information using embedded sensors in smart devices," *IEEE Netw.*, vol. 32, no. 4, pp. 8–14, Jul./Aug. 2018.

[33] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: Privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1868–1878, Dec. 2017.

[34] J. L. Z. Cai, M. Yan, and Y. Li, "Using crowdsourced data in location-based social networks to explore influence maximization," in *Proc. IEEE INFOCOM-35th Annu. IEEE Int. Conf. Comput. Commun.*, Apr. 2016, pp. 1–9.

[35] Z. Xiong, Z. Cai, Q. Han, A. Alrawais, and W. Li, "ADGAN: Protect your location privacy in camera data of auto-driving vehicles," *IEEE Trans. Ind. Informat.*, early access, Oct. 20, 2020, doi: 10.1109/TII.2020.3032352.

[36] X. Jiang, F. R. Yu, T. Song, Z. Ma, Y. Song, and D. Zhu, "Blockchain-enabled cross-domain object detection for autonomous driving: A model sharing approach," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3681–3692, May 2020, doi: 10.1109/JIOT.2020.2967788.

ZHENGUO DING is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Nanjing University of Science and Technology, China. His research interests include penetration testing, software testing, and vulnerability mining.

SITONG LIU received the B.Sc. degree from the Nanjing University of Science and Technology, Nanjing, China, in 2018, where he is currently pursuing the Ph.D. degree. His research interests include object detection, deep learning attack, and defense.

MIN LI received the Ph.D. degree in biomedical engineering from Chongqing University, in 2014. From 2011 to 2013, she was a Visiting Ph.D. Student with the Department of Radiation Oncology, The University of Texas MD Anderson Cancer Center, Houston, TX, USA. She is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. Her current research interests include image processing, analysis, and visualization.

ZHICHAO LIAN (Member, IEEE) received the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2013. From 2012 to 2014, he was a Postdoctoral Associate with the Department of Statistics, Yale University. He is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include image processing, pattern recognition, and the IoT.

HUI XU received the B.S. degree in electronics science and technology and the M.S. degree in signal and information processing from the Nanjing University of Posts and Telecommunications Nanjing, China, in 2008 and 2011, respectively. He is currently a Staff Engineer with Samsung Electronics (China) Research and Development Centre. His research interests include network security, cloud computing, and high availability infrastructure.