Intelligent Person Monitoring system of the UHV Power Infrastructure Line Construction site based on Deep Neural Networks

Keliang Zhu
State Grid Anhui Electric Power Co, Ltd. Construction Company, Hefei, Anhui, China
524104473@qq.com

Xuemei Shi
State Grid Anhui Electric Power Co, Ltd. Construction Company, Hefei, Anhui, China
921156702@qq.com

Tianzhong Zhang
State Grid Anhui Electric Power Co, Ltd. Hefei, Anhui, China
738570760@qq.com

Huasong Song
State Grid Anhui Electric Power Co, Ltd. Construction Company, Hefei, Anhui, China
1194025698@qq.com

Jinlin Xu
Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei, Anhui, China
jlxu@hfcas.ac.cn

Liangfeng Chen*
Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei, Anhui, China
quinear@hfcas.ac.cn

ABSTRACT

Recently, many types of workers are required for the ultra-high voltage (UHV) power line construction sites in China. For safety, each worker's real-time status should be monitored. The key point of this monitoring system is identifying and tracking each worker accurately and quickly, which is the special multi-objects tracking (MOT) problem at the large open scene. In this paper, an intelligent person monitoring system with sensors composed of a camera and the ultra-wide band (UWB) radars is proposed. The proposed MOT method is composed of two parts. The one is the method of detecting and identifying persons, which is composed of an image person recognition method that is a Faster R-CNN combined with a re-ID branch, a UWB worker localization method, and a person association method. And the other is tracking these persons, which is composed of an person-track association algorithm by a similarity matrix, and a tracking method by the Kalman Filter. The experiments illustrate that our method can monitor multi-persons accurately and quickly in real time.

CCS CONCEPTS

• **Information systems**; • **Information systems applications**; • **Computing platforms**;

KEYWORDS

ultra-wide band (UWB), Faster R-CNN, the intersection over union (IoU), ultra-high voltage engineering(UAV), similarity metric

*Fund support: Science and Technology Projects of State Grid Corporation of China (Grant No. B3120A190005).

APIT 2023, February 09–11, 2023, Ho Chi Minh City, Vietnam

ACM ISBN 978-1-4503-9950-0/23/02…$15.00
https://doi.org/10.1145/3588155.3588175

ACM Reference Format:
Keliang Zhu, Xuemei Shi, Tianzhong Zhang, Huasong Song, Jinlin Xu, and Liangfeng Chen. 2023. Intelligent Person Monitoring system of the UHV Power Infrastructure Line Construction site based on Deep Neural Networks. In *2023 5th Asia Pacific Information Technology Conference (APIT) (APIT 2023), February 09–11, 2023, Ho Chi Minh City, Vietnam.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3588155.3588175

1 INTRODUCTION

Recently, the pace of power infrastructure construction has been significantly accelerated in China, especially the ultra-high voltage engineering. Since, the safety guarantee of workers at power infrastructure construction site has attacked widespread attention [1]. In this paper, we focus on the special safety issues in the early stage of the line construction site, such as the tower foundation construction etc. In this scene, workers are managed by a supervision person without any intelligent supervision equipment due to no electricity. Since the security risks are completely unpredictable, the most serious person security accidents can't be prevented so as to cause serious economic losses.

To solve these problems, we established an intelligent person monitoring system with one camera and the ultra-wide band (UWB) radar that is a type of low-power consumption localization equipment. In the system, all collected raw site data is transferred to the cloud server to analyse, and then the analysis results are transferred to the supervision person's mobile phone to help him to manage workers. The core of the system is how to recognize and track multi-objects with mono camera and the UWB equipment accurately.

In this paper, we identify and track all persons by the frame images captured by the mono camera, and improve the persons' recognizing and tracking accuracy by the UWB. First, we extract feature maps from the frame images. Second, we take Faster R-CNN to detect the targets and get the category and coordinate. A re-ID branch is combined with Faster R-CNN to recognize the persons. In re-ID branch, we further extract the re-ID features and get the ID annotation of each person, and improve the re-ID branch accuracy with the UWB recognition results. Based on the camera, we can get information of each person, such as the coordinate, the virtual ID,

the class. Third, in the tracking network, we establish an person-track similarity matrix, and its element is calculated by a mixed metric distance including the coordinate, and the intersection over union (IoU) info. And finally, persons' tracks are improved by the Kalman Filter.

Compared to the existing works, the main contributions of this paper are as follows.

- A novel network structure is proposed to detect and identify persons, which is combined of the Faster R-CNN and re-ID network branch.
- A novel tracking method based on the person-track similarity metric with the mixed distance is proposed.

The rest of this paper is organized as follows. Section 2 introduces the current related works of the MOT. Section 3 exhibits our multi-objects tracking method. Experiments are given in Section 4 to illustrate performances of our method. Finally, some conclusions and future works are given in Section 5.

2 RELATED WORDS

From the perspective of image acquisition, cameras and stereo RGB-D cameras are widely used in multi-object recognition and tracking. For example, Jiang et al. [2] presented a multiple pedestrian tracking method for videos captured by a fixed camera. Zimmermann et al. [3] proposed a hand pose detection method based on cameras. These methods that consider RGB-only come with some limitations. They struggle with postures, occlusions and motion blur. The reason is that a camera only makes observations in the 2D image plane and cannot directly measure the distance to objects. Thus, some researchers used hybrid device to achieve multi-object tracking. Sun et al. [4] reduced the influence of electromagnetic interference on sensor data accuracy, while it's not needing in the line site with no power in this paper. Zhao et al. [5] developed a framework for moving vehicle detecting, tracking and geolocating based on a camera, a GPS receiver and inertial measurement unit sensors. Sridhar et al. [6] used five cameras and an additional depth sensor to demonstrate real-time hand pose estimation. Li et al. [7] studied 3D bounding box tracking with stereo RGB-D cameras. Dieterle et al. [8] proposed a multi-object tracking method using a stereo RGB-D camera and a laser range finder. The stereo RGB-D cameras can perceive 3D depth directly, but is not possible in many cases [9]-[12]. Li et al.[13] proposed a novel data fusion method but it's not applicable to the RGB-D data fusion scene.

For the videos from cameras or RGB-D cameras, some researchers presented many methods to achieve multi-object recognition and tracking. Early methods track objects based on correlation filters, such as MOSSE [14], MCPF [15], KCF [16], Kalman filter [17] and Staple [18]. In recent years, techniques based on deep learning have achieved excellent results in multi-object recognition and tracking. MOTDT [19] is a real-time multiple people tracking method that combines Kalman filter with deep neural network. Xu et al. [20] proposed DeepMOT based on a bidirectional recurrent network to train multiple target trackers. Chen et al. [21] proposed EDMT to detect and track multiple targets. Especially, the multi-object recognition and tracking reaped many of the benefits from the success of convolutional representation. Mahmoudi et al. [22] proposed a CNN-based method (CNNMTT) for multi-object tracking. Girshick et al. [23] proposed regions of convolutional neural network features (R-CNN). The method gets region proposals by selective search and uses CNN features to classify the region of interest (ROI). Girshick [24] further proposed Fast R-CNN with ROI pooling layer. The region proposals are projected on the feature map by ROI pooling, which decreases the processing time. After that, Ren et al. [25] proposed Faster R-CNN. Faster R-CNN integrates the Fast R-CNN and region proposal network (RPN). The method gets region proposals from feature maps generated by RPN instead of selective search. Mask R-CNN [26] is further proposed based on Faster R-CNN. It contains three modules: Faster R-CNN, ROI Align and full convolution network (FCN). Furthermore, Track R-CNN [27] extended the Mask R-CNN and added an identity embedding net-work to extract re-ID features. Tracktor [28] is also an adaptation of the Faster R-CNN to the multi-object tracking task. It uses a Faster R-CNN to detect targets and to follow the detected targets in the consecutive frames. Baisa [29] posed a novel online multi-object visual tracker using a Gaussian mixture Probability Hypothesis Density (GM-PHD) filter and deep appearance learning. The GM-PHD filter is to estimate the states and cardinality of time-varying number of objects, and the deep appearance learning is to perform estimates-to-tracks data association for target labeling as well as formulate an augmented likelihood and then inte-grate into the update step of the GM-PHD filter.

3 METHODOLOGY

3.1 Overview

In the line site, the camera is fixed at any corner of the site to monitor the whole site, and the four UWB base stations are deployed at each corner of the site to locate each worker. Since, the tracking method should fuse data from multi-sensors, and the tracking process (see Figure 1). First, all persons should be recognized by the image and the UWB. While, the boxes in the image are recognized by the Fast R-CNN with the re-ID branch, and the boxes formed by the UWB are drawn by each person's position and its height and width. Second, a person association algorithm is proposed to localize each person with the above two boxes, which is the box align algorithm in the Figure 1. Third, each person's track can be calculated by the Kalman Filter (KF) with its positions before time t. Fourth, a person-track data association algorithm that is defined as the similarity matrix is proposed to match the person to his pre-track. Finally, refreshing each person's track. And the core algorithms are illustrated in details.

3.2 Multi-Persons Recognition

3.2.1 Recognition with the Image data. The persons in the image are recognized by the Fast R-CNN with a re-ID branch (see Figure 2). The method is composed by three parts: a CNN backbone, a detection branch and a re-ID branch. The backbone network is responsible for extracting feature maps from images. The detection branch is used to get the category and coordinate of each person in the images. The re-ID branch identifies each detection object in the same class which is person in this paper.

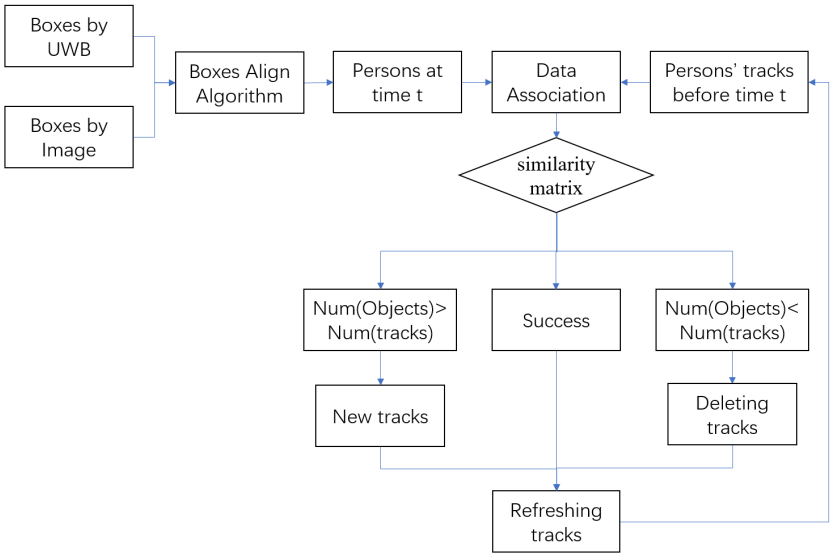

Figure 1: the MOT method.

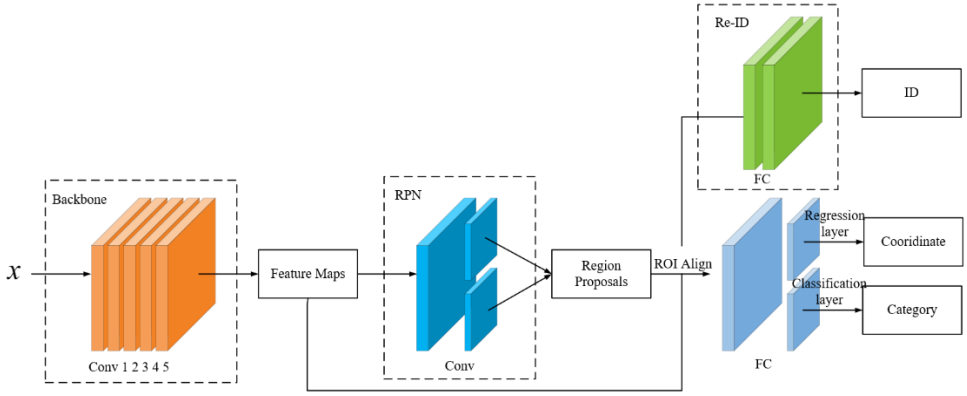

Figure 2: Overview of multi-objects recognition.

Backbone Network. We adopt VGG-19 [30] as backbone network to extract image features. The network consists of sixteen convolutional layers and four max-pooling layers (see Table 1). The inputs of backbone network are images from video frames. Denote the input size as $h \times w$, in which h and w are the height and width of the image, respectively. The convolutional layers involve a filter $W \in \mathbb{R}^{3\times3}$, where 3×3 is the filter size. The convolutional operation maps the input image to a latent feature map $c = \sigma(W*x + b)$, where $\sigma(\cdot)$ is the activation function ReLU, x is the input image, $*$ is the convolutional operation, and b is the corresponding bias. The convolutional layers are followed by max-pooling layers with pooling size of 2×2. Different from original VGG-19, we abandon three fully-connected layers to speed up the feature extraction.

Detection Branch. The detection branch is a common Faster R-CNN architecture that consists of a RPN, followed by classification layer and regression layer. This RPN takes the feature maps from the backbone network as input and outputs rectangular region proposals. Then we align the extracted feature maps with the region proposals. We classify the regions by classification layer and locate the coordinates by regression layer. The output of the classification layer is the probability distribution of each candidate region, which is denoted as $p = (p_i, p_i^*)$. p_i is the probability that the candidate region is the target, and p_i^* is the probability that the candidate

Table 1: Details of backbone network.

Kernel size	Kernel size	Output
Conv 1.1, 1.2, Max-pool	3×3	h/2×w/2×64
Conv 2.1, 2.2, Max-pool	3×3	h/4×w/4×128
Conv 3.1, 3.2, 3.3, 3.4, Max-pool	3×3	h/8×w/8×256
Conv 4.1, 4.2, 4.3, 4.4, Max-pool	3×3	h/16×w/16×512
Conv 5.1, 5.2, 5.3, 5.4	3×3	h/16×w/16×512

region is not the target. The output of regression layer is the predicted coordinate $t_i = (t_x, t_y, t_w, t_h)$. The classification loss is a cross entropy loss and the regression loss is a smooth L1 loss [20]. The joint loss function is shown as below:

$$L_{detection}\left(p_i, t_i\right) = \frac{1}{N}\sum_{i=1}^{N} L_{cls}\left(p_i, p_i^*\right) + \lambda\frac{1}{N}\sum_{i=1}^{N} p_i^* L_{reg}\left(t_i, t_i^*\right) \quad (1)$$

where N is the number of candidate regions, λ is the corresponding coefficient. L_{cls} is the classification loss and L_{reg} is the regression loss, denoted as:

$$L_{cls}\left(p_i, p_i^*\right) = -\log\left[p_i^* p_i + \left(1 - p_i^*\right)\left(1 - p_i\right)\right] \quad (2)$$

$$L_{reg}\left(t_i, t_i^*\right) = R\left(t_i - t_i^*\right) \quad (3)$$

where R is the smooth L1 loss function, t_i^* is the true coordinate of the targets.

To accommodate for this difference, we average the detection loss $L_{detection}$ from both video frames while training our detection branch.

Re-ID Branch. The re-ID branch aims to identify an ID of each person. We use two fully-connected layers to extract re-ID features for each target in the same class which is the person. Then we map the feature vector to an ID distribution vector $P = \{p(k), k \in [1, K]\}$. The re-ID loss is a cross entropy loss, which is shown as bellow:

$$L_{identity} = -\sum_{k=1}^{K} L\left(k\right)\log\left(p\left(k\right)\right) \quad (4)$$

where K is the number of IDs for each person, $L(k)$ is the one-hot representation of the ID label of a real person and his related to the UWB's ID, and $p(k)$ is defined as follow.

$$p\left(k\right) = norm\left(\frac{Box_{img}^k \cap Box_{UWB}^k}{Box_{img}^k \cup Box_{UWB}^k}\right) * \frac{1}{K} \quad (5)$$

where Box_{img}^k is the k-th person's box detected by the detection branch, and Box_{UWB}^k is the box drawn by the same person's position calculated by the UWB, and norm is a normalization function to ensure $\sum_{k=1}^{K} p(k) = 1$. While, if the Box_{UWB}^k is not exist, it's defined as Box_{img}^k.

3.2.2 Recognition with the UWB data. Usually, each worker that is a special identity person is required to equip his unique UWB receiver at his middle waist position. While each worker's location is calculated by the following formula.

$$\begin{cases} \sqrt{(x_k - x_1)^2 + (y_k - y_1)^2 + (z_k - z_1)^2} = d_1 \\ \sqrt{(x_k - x_2)^2 + (y_k - y_2)^2 + (z_k - z_2)^2} = d_2 \\ \sqrt{(x_k - x_3)^2 + (y_k - y_3)^2 + (z_k - z_3)^2} = d_3 \end{cases} \quad (6)$$

where (x_k, y_k) is the worker's location that is unknown and calculated by the least square algorithm, z_k is set equal to one meter, (x_i, y_i, z_i), $i = 1, 2, 3$ is the UWB station's location fixed at each corner of the site, $d_i, i = 1, 2, 3$ is the distance between the worker and each UWB station, which is a measured value by the UWB. To decrease the location error, d_i with a smaller value is selected from the four UWB station.

In the camera's perspective, the k-th worker's box is drawn as $(x_k - W/2, y_k - H/2), (x_k + W/2, y_k + H/2)$, where the first coordinate is the top left corner of the box, and the last is the lower right corner, W is a worker's width and set to 0.5m, H is a his height that is a different known value.

3.2.3 Multi-Objects Association with the mixed results. After each person is recognized respectively by the UWB and the image, a box align algorithm should be proposed. To associate data from different sensors, we use the spatial coordinate transformation [8] method to transform the UWB localization to the camera coordinate. And the algin steps are shown as follows.

Algorithm 1 Person Box Align Algorithm

Inputs: Box_{img}^K, Box_{UWB}^M // boxes detected by the image and the UWB, K, M is the total number
Step 1:
$N = max(Box_{img}^K, Box_{UWB}^M)$ while i<N, do:
Step 2:
$Box_i = Box_{img}^i \cap Box_{UWB}^i$ // when one of Box_{img}^k and Box_{UWB}^m is not exist, $Box_i = Box_{img}^i$ *or* Box_{UWB}^i
Step 3:
the k-th person's coordinate is
<?TeX $\left(x_{Box_i}, y_{Box_i}\right) = center\left(Box_i\right)$?>
Outputs: each person's box and his coordinate; the total number N.

3.3 Data associated by the Similarity Matrix

Ideally, we get the track by lining the persons with same IDs calculated by the re-ID branch. However, the appearance of the persons can change rapidly over time due to pose changes, occlusion and motion blur, and the tracks of different persons can intersect while crossing each other. Thus, some persons may have no IDs or wrong IDs, which leads the track to be disconnected or wrong. While, UWB tag with person's identify info can associated to the virtual IDs. So, an accurate similarity metric distance between the person and his tracks should be designed properly. For convenience, some definitions are described as follows. $D = \{Box_1^t, Box_2^t, \cdots, Box_N^t\}$ denotes all detected persons by the Algorithm 1. $T = \{Tr_1^{t-1}, Tr_2^{t-1}, \cdots, Tr_M^{t-1}\}$ denotes the tracks in $(t-1)$-th video frame, M is the number of tracks. Tr_j^{t-1} denotes the j-th track drawn with person $Box_i^k (k \in [1, t-1])$.

We apply intersection over union (IoU) and distances to measure the similarity between the current persons and tracks.

$$d_{ij} = dL_{ij}/IoU_{ij} \quad (7)$$

where dL_{ij} and IoU_{ij} are defined as follows:

$$dL_{ij} = \sqrt{\left(x_{Box_i^t} - x_{Tr_j^{t-1}}\right)^2 + \left(y_{Box_i^t} - y_{Tr_j^{t-1}}\right)^2} \quad (8)$$

$$IoU_{ij} = (1-\alpha)\frac{Box_i^t \cap TrB_j^{t-1}}{Box_i^t TrB_j^{t-1}} + \alpha\frac{Box_i^t \cap TrB_j^{t-1}}{Box_i^t TrB_j^{t-1}} \quad (9)$$

where TrB_j^{t-1} is the j-th person's box expanded by his width and height in time t-1. The expanded method is the same as drawing the box by the UWB.

Based on the mixed similarity metric distance (Formula 5), the similarities between N persons and M tracks are calculated by matrix S_{NM}.

$$S_{NM} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1M} \\ d_{21} & d_{22} & \cdots & d_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & \cdots & d_{NM} \end{bmatrix} \tag{10}$$

where row i denotes the similarities between the current person Box_i^t and the person in the final frame of all tracks. Likewise, column j denotes the similarities between the person in the final frame of j-th track and all persons.

For each row, the minimum distance value d_{ij} is selected for the j-th track. Meanwhile, for each column, it's also the minimum for the Box_i^t. Otherwise, the tracking is abandoned. When N<M, recoding the losing persons until losing three moments. If N>M, tracking the new persons.

3.4 Refreshing tracks by KF

After each person and his track are associated, the tracks should be refreshed by the Kalman Filter (KF) shown as follows.

$$\begin{cases} x_t = Ax_{t-1} + Bu_{t-1} + w_{t-1}, \quad p\,(w) \sim N\,(0, Q) \\ z_t = Hx_t + v_t, \quad p\,(v) \sim N\,(0, R) \end{cases} \tag{11}$$

where x_t is a estimating state vector at time t, including the k-th person's coordinate and his velocity, u_{t-1} is a control vector that is set to zero vector in this paper, z_t is the measurement vector, and its coordinate is the centre of Box_i calculated by the Algorithm 1, A, B, H is a parameter matrix, and the random variable w_{t-1} represents the process noise, v_t represents the measurement noise.

In the Time Update (Prediction Stage)

$$\begin{cases} \hat{x}_t^- = A\hat{x}_{t-1} \\ P_t^- = AP_{t-1}A^T + Q \end{cases} \tag{12}$$

In the Measurement Update (Correction Stage)

$$\begin{cases} K_t = P_t^- H^T \left(HP_t^- H^T + R\right)^{-1} \\ \hat{x}_t = \hat{x}_t^- + K_t\left(z_t - H\hat{x}_t^-\right) \\ P_t = (I - K_t H)\, P_t^- \end{cases} \tag{13}$$

4 EXPERIMENTS AND RESULTS

4.1 Implementation Details

The experiments are implemented in an Ubuntu 16.04 computer with an Intel(R) Xeon(R) CPU, 32 GB RAM, 500 GB SSD, 2 TB HDD and a 12 GB NVIDIA TITAN Xp GPU. The version of TensorFlow-GPU is 1.6.0 and CUDA is 9.0. The backbone network is pre-trained on the COCO dataset [31]. Then we further trained detection branch and re-ID branch. The batch size is set to be 10. The epoch number is set to be 30. We apply Adam optimizer [32] and the learning rate is 0.004. The coefficient $\boldsymbol{\lambda}$ is set to be 1, and the coefficient $\boldsymbol{\alpha}$ is set to be 0.5.

4.2 Datasets and Metrics

We conduct our experiments on MOT16 and MOT17 datasets [33], which are representative benchmarks. The MOT16 dataset contains 14 real-world video sequences for both dynamic and static scenes, in which are seven sequences for training dataset and seven sequences for test dataset. Each dataset has four sequences with moving cameras and three sequences with static cameras. The MOT16 dataset provides 564228 manually annotated bound boxes and their identity annotations. The MOT16 dataset provides public detections of the DPM detector [34]. The MOT17 dataset has the same video sequences as the MOT16, but it provides public detections of three detectors that are produced by DPM, Faster-RCNN and SDP[35].

The metrics used for evaluating our method are the multi-objects tracking accuracy (MOTA), the multi-objects tracking precision (MOTP), the mostly tracked targets (MT), the mostly lost targets (ML), the number of false positive (FP), the number of false negative (FN) and the number of identity switches (IDSW). MOTA is the overall tracking accuracy in terms of false positives, false negatives and identity switches, which gives a measure of the tracker's performance at detecting objects as well as keeping track of their tracks. MOTP is the overall tracking precision in terms of bounding box overlap between ground truth and tracked location, which shows the ability of the tracker to estimate precise objects positions. MT is the percentage of mostly tracked targets that are successfully tracked for more than 80% of its ground truth boxes. ML is the percentage of mostly lost targets that fail to be tracked for more than 20% of its ground truth boxes. FP is the number of false detections. FN is the number of miss detections. IDSW is the number of times the given identity of a ground truth track changes.

4.3 Analysis of Results

4.3.1 Results with the public datasets. For a comparison with other methods, we trained our model separately using the training dataset from MOT16 and MOT17, and separately applied the model on the MOT16 test dataset and MOT17 test dataset. Table 2 and Table 3 compare the tracking results of some methods mentioned in this paper on MOT16 and MOT17 test dataset.

From Table 2 and Table 3 , we can see that our method gets almost the best results on both MOT16 and MOT 17. In Table 2, our MOTA metric is 54.8 that is slightly lower than CNNMTT. The MOTP metric achieves 79.0 and it is the best among all results. The MT metric is 23.4%. The ML metric is 19.1%. The FP metric is 5104. The FN metric is 86245. The IDSW metric is 481. In Table 3, the MOTA metric achieves 55.4 and exceeds DeepMOT by 1.7. The MOTP metric achieves 78.1 and it is slightly higher than Tracktor. The MT metric is 23.4% and exceeds EDMT by 1.8%. The ML metric is 35.7%. The FP metric is 22213. The FN metric is 232659. The IDSW metric is 1407. As a result, our method has good performance in terms of MOTA, MOTP, MT, ML, FP, FN and IDSW.

4.3.2 Results with the data in the site. Based on the real data, in the case of using the camera and the UWB, the evaluating metrics are significantly better than the values in the case of using the camera only. The MOTA in the pre-case is upper 30% than the last, the MOTP is upper 12%, the MT is upper 80%, the ML is lower 73%, the FP is lower 7.6%, the FN is lower 8%, and the IDSW is lower 80%. Obviously, in the pre-case, the accuracy of the re-ID branch is improved Significantly.

Table 2: Comparisons of tracking results on the MOT16.

Methods	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓	IDSW↓
KCF	48.8	75.7	15.8%	38.1%	5875	86567	906
MOTDT	47.6	74.8	15.2%	38.3%	9253	85431	792
DeepMOT	54.8	77.5	19.1%	37.0%	**2955**	78765	645
EDMT	45.3	75.9	17.0%	39.9%	11122	87890	639
CNNMTT	**65.2**	78.4	**32.4%**	21.3%	6578	**55896**	946
Tracktor	54.4	78.2	19.0%	36.9%	3280	79149	682
Ours	54.8	**79.0**	23.4%	**19.1%**	5104	86245	**481**

Table 3: Comparisons of tracking results on the MOT17.

Methods	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓	IDSW↓
MOTDT	50.9	76.6	17.5%	35.7%	24069	250768	2474
DeepMOT	53.7	77.2	19.4%	36.6%	11731	247447	1947
EDMT	50.0	77.3	21.6%	36.3%	32279	247297	2264
Tracktor	53.5	78.0	19.5%	36.6%	12201	248047	2072
Ours	55.4	78.1	23.4%	35.7%	22213	232659	1407

5 CONCLUSIONS

In this paper, we propose a hybrid method for intelligent person monitoring. Different from existing methods, we use a camera and the UWB radars to achieve high accuracy. Our recognition method improves the image re-ID branch accuracy with the UWB results, and solving the losing person problem in the image. The person-track association similarity matrix is improved, and its element similarity metric is composed of the IoU distance and the location distance. At last, we improve the tracking accuracy by the KF. Experiments on representative benchmarks MOT16 and MOT17 demonstrate the superiority of our method in terms of both efficiency and accuracy of similar target recognition and tracking. In reality site, the average tracking accuracy can reach to 0.2m, and the ID changing frequency levels off to 5%. While, this case must be under the condition that each worker should wear UWB properly.

Data Availability

The data used to support the findings of this study are public and included within the article.

Conflicts of Interest

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

ACKNOWLEDGMENTS

This work was supported by Science and Technology Projects of State Grid Corporation of China (Grant No. B3120A190005).

REFERENCES

[1] Wei Sun, Kang Wei, Zhi Liu, Qiyue Li, Xiaobing Xu, "Linear Quadratic Gaussian Control for Wireless Communication Reliability for a Mobile Monitoring Robot in a UHV Power Substation," *IEEE Systems Journal*, pp. 1–11,2022

[2] Z. Jiang and D. Q. Huynh, "Multiple pedestrian tracking from videos in an interacting multiple model framework," *IEEE Trans. on Image Process*, vol. 27, pp. 1361–1375, 2018.

[3] C. Zimmermann and T. Brox, "Learning to estimate 3d hand pose from single RGB images," *Proc. of 2017 IEEE Int. Conf. on Computer Vision (ICCV), Venice, Italy*, pp. 4903–4911, 2017.

[4] Wei Sun, Xiaojing Yuan, Jianping Wang *et al.*, "End-to-End Data Delivery Reliability Model for Estimating and Optimizing the Link Quality of Industrial WSNs," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1127–1137, 2018

[5] X. Zhao, F. Pu, Z. Wang, H. Chen and Z. Xu, "Detection, tracking, and geolocation of moving vehicle from UAV using camera," *IEEE Access*, vol. 7, pp. 101160–101170, 2019.

[6] S. Sridhar, A. Oulasvirta and C. Theobalt, "Interactive markerless articulated hand motion tracking using RGB and depth data," *Proc. of 2013 IEEE Int. Conf. on Computer Vision (ICCV), Sydney, Australia*, pp. 2456–2463, 2013.

[7] P. Li, T. Qin and S. Shen, "Stereo vision–based semantic 3d object and ego-motion tracking for autonomous driving," *Proc. of European Conf. on Computer Vision (ECCV)*, pp. 646–661, 2018.

[8] T. Dieterle, F. Particke, L. Patino-Studencki and J. Thielecke, "Sensor data fusion of LIDAR with stereo RGB-D camera for object tracking," *2017 IEEE Sensors, Glasgow, UK*, pp. 1–3, 2017.

[9] H. N. Hu, Q. Z. Cai, D. Wang, J. Lin, M. Sun *et al.*, "Joint 3D vehicle detection and tracking," *Proc. of 2019 IEEE Int. Conf. on Computer Vision (ICCV), Seoul, Korea*, pp. 5390–5399, 2019.

[10] R. Mur-Artal and J. D. Tardos, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. on Robotics*, vol. 33, No. 5, pp. 1255–1262, 2017.

[11] C. Jing, J. Potgieter, F. Noble and R. Wang, "A comparison and analysis of RGB-D cameras' depth performance for robotics application," *Proc. of 2017 24th Int. Conf. on Mechatronics and Machine Vision in Practice (M2VIP), Auckland, New Zealand*, pp. 1–6, 2017.

[12] K. Chen, Y. K. Lai and S. M. Hu, "3D Indoor scene modeling from RGB-D data: A survey," *Comput. Vis. Media*, vol. 1, pp. 267–278, 2015.

[13] Q Li, T Cao, W Sun *et al.*, "An Optimal Uplink Scheduling in Heterogeneous PLC and LTE Communication for Delay-aware Smart Grid Applications," *Mobile Networks and Applications*, vol.26, pp.1–14, 2021.

[14] K. Han, "Image object tracking based on temporal context and MOSSE," *Cluster Comput*, vol. 20, pp. 1259–1269, 2017.

[15] T. Zhang, C. Xu and M. H. Yang, "Multi-task correlation particle filter for robust object tracking," *Proc. of 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA*, pp. 4819–4827, 2017.

[16] J. F. Henriques, R. Caseiro, P. Martins and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 37, No. 3, pp. 583–596, 2015.

[17] S. E. Li, G. Li, J. Yu, C. Liu, B. Cheng *et al.*, "Kalman filter-based tracking of moving objects using linear ultrasonic sensor array for road vehicles," *Mech. Syst. Signal Proc.*, vol. 98, pp. 173–189, 2018.

[18] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik and P. H. Torr, "Staple: Complementary learners for real-time tracking," *Proc. of 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA*, pp. 1401–1409, 2016.

[19] L. Chen, H. Ai, Z. Zhuang and C. Shang, "Real-time multiple people tracking with deeply learned candidate selection and person re-identification," *Proc. of*

2018 IEEE Int. Conf. on Multimedia and Expo (ICME), San Diego, CA, USA, pp. 1–6, 2018.

[20] Y. Xu, Y. Ban, X. Alameda-Pineda and R. Horaud, "DeepMOT: A differentiable framework for training multiple object trackers," ArXiv Preprint ArXiv:1906.06618, 2019.

[21] J. Chen, H. Sheng, Y. Zhang and Z. Xiong, "Enhancing detection model for multiple hypothesis tracking," *Proc. of 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii*, pp. 2143–2152, 2017.

[22] N. S. Mahmoudi, M. Ahadi and M. Rahmati, "Multi-object tracking using CNN-based features: CNNMTT," *Multimed. Tools Appl.*, vol. 78, pp. 7077–7096, 2019.

[23] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proc. of 2014 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA*, pp. 580–587, 2014.

[24] R. Girshick, "Fast R-CNN," Proc. of 2015 IEEE Int. Conf. on Computer Vision (ICCV), Santiago, Chile, pp. 1440–1448, 2015.

[25] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 39, No. 6, pp. 1137–1149, 2017.

[26] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," *Proc. of 2017 IEEE Int. Conf. on Computer Vision (ICCV), Venice, Italy*, pp. 22–29, 2017.

[27] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B. B. G. Sekar *et al.*, "MOTS: Multi-object tracking and segmentation," *Proc. of 2019 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA*, pp. 7942–7951, 2019.

[28] P. Bergmann, T. Meinhardt and L. Leal-Taixe, "Tracking without bells and whistles," *Proc. of 2019 IEEE Int. Conf. on Computer Vision (ICCV), Seoul, Korea*, pp. 941–951, 2019.

[29] Baisa Nathanael L. , "Occlusion-robust online multi-object visual tracking using a GM-PHD filter with CNN-based re-identification," *Journal of Visual Communication and Image Representation*, vol. 80, 2021

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," ArXiv Preprint ArXiv:1409.1556, 2014.

[31] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona *et al.*, "Microsoft coco: Common objects in context," *Proc. of European Conf. on Computer Vision (ECCV), Zurich, Switzerland*, pp. 740–755, 2014.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," ArXiv Preprint ArXiv:1412.6980, 2014.

[33] A. Milan, L. Leal-Taixé, I. Reid, S. Roth and K. Schindler, "MOT16: A benchmark for multi-object tracking," ArXiv Preprint ArXiv:1603.00831, 2016.

[34] P. Felzenszwalb, R. Girshick, D. McAllester and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, No. 9, pp. 1627–1645, 2010.

[35] F. Yang, W. Choi and Y. Lin, "Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers," *Proc. of 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA*, pp. 2129-2137, 2016.