ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

Research on artificial intelligence target tracking algorithm based on computer vision

Xiaokai Jiang*, Xuewen Ding, Chunyu Liu, Yuan Zhang, Shaosai Wang

School of Electronic Engineering, Tianjin University of Technology and Education, Tianjin, China
**Corresponding author: 2506901369@qq.com*

Abstract: *In today's society, computer vision technology has become an important part of the field of artificial intelligence, and plays a key role in many practical application scenarios. This paper mainly discusses the research progress and application of target tracking algorithm based on computer vision. Firstly, the basic concept and technical background of target tracking are summarized, including key technical links such as target detection, feature extraction and motion prediction. Then, several mainstream target tracking methods, such as correlation filter tracking, deep learning tracking and model-based tracking, are analyzed, and their advantages and disadvantages are compared. Additionally, this paper accords prominence to the utilization of deep learning technology in the domain of target tracking, presenting a selection of sophisticated neural network-based tracking algorithms and assessing their respective performances. In addition, some improvement measures and solutions are proposed to solve the challenges of existing tracking algorithms, such as occlusion, illumination change, scale change, etc. Finally, this paper compares the performance of different tracking algorithms through experiments.*

Keywords: *Computer vision; Target tracking; Deep learning; Feature matching; Algorithm evaluation*

1. Introduction

Object tracking technology, being a pivotal research avenue within the realm of computer vision, endeavors to detect and recognize mobile objects for the purpose of analyzing their motion trajectories and state alterations across video sequences. As computer hardware performance enhances and deep learning algorithms achieve breakthroughs, target tracking has emerged as a pivotal component in a myriad of intelligent applications, including intelligent monitoring, automatic driving, drone monitoring and human-computer interaction scenarios. Target tracking requires not only efficient and stable target positioning, but also rapid response in complex dynamic environment, which puts forward strict requirements for algorithm design.

The process of target tracking can be divided into several key links, in which target detection, feature extraction and motion prediction are the most basic and key technical links. First of all, target detection is the starting point of target tracking, and its task is to identify the position of the target object in each frame image. Previously, traditional object detection algorithms heavily relied on features that were crafted by human design. However, in recent times, convolutional neural network (CNN)-based methodologies, including Faster R-CNN and YOLO (You Only Look Once), have gained prominence, have rapidly improved the accuracy and real-time detection of objects. These deep learning frameworks can deal with object detection tasks in complex scenarios more effectively by automatically learning features in a large number of sample data.

Secondly, feature extraction plays a crucial role in target tracking. The purpose of feature extraction is to convert the appearance information of the target into a low-dimensional feature space, so that the subsequent tracking algorithm can describe and match the target. Features can be color histograms, edge features, texture features, or even high-level abstract features obtained through deep learning models. Efficient feature extraction significantly fortifies the robustness of targets amidst varying lighting conditions, viewing angles, and background shifts. In recent years, deep feature learning has emerged as the predominant approach. As deep learning technology advances, numerous target tracking algorithms are incorporating convolutional neural networks to refine the feature extraction process, thereby enhancing the precision of tracking.

Ultimately, motion prediction constitutes an integral segment within the target tracking process. The

ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

anticipation of the target's trajectory in both temporal and spatial dimensions contributes to enhancing the stability of tracking, particularly in scenarios involving brief occlusions or swift movements of the target. Motion prediction usually uses kinematic knowledge based on physical models, or machine learning techniques to capture the laws of target motion. Common motion models include uniform motion model and uniformly accelerated motion model, while modern methods combine time series prediction based on long short-term memory (LSTM) network to deal with target motion in dynamic scenes more effectively.

Although target tracking technology has made remarkable progress recently, it still faces many challenges in complex practical applications. Certain factors, including occlusion, scene interference, target deformation, and dynamic background changes, can adversely impact the performance of tracking algorithms. Therefore, how to design a target tracking algorithm that can adapt to changing environment and effectively deal with various challenges has become an urgent problem for researchers.

In short, object tracking technology is a multidisciplinary research field, integrating image processing, machine learning, kinematics and many other knowledge. This paper will systematically discuss the research progress of artificial intelligence object tracking algorithms based on computer vision, analyze the advantages and disadvantages of current mainstream technologies, and their performance in practical applications, so as to provide reference and inspiration for future research. By gaining a deeper understanding of key techniques such as target detection, feature extraction and motion prediction, we look forward to contributing to improving the accuracy and efficiency of target tracking[1-2].

2. Research status

2.1 Present situation of traditional target detection

The basic process of the traditional object detection method includes: first, select the candidate region in the input image; Secondly, feature extraction is carried out for these regions. Finally, the background or target is classified by a pre-trained classifier. At present, the usual way to generate candidate regions is the sliding window technique, which slides sequentially over the detection image to select regions of interest. Owing to the extensive generation of candidate regions throughout the process, the computational requirement escalates. To mitigate the issue of an overabundance of candidate boxes, the BING algorithm introduced in 2012 offers a solution by identifying more targets within a reduced number of areas, thereby markedly lessening the computational load and accelerating the detection process.

Regarding feature extraction, traditional algorithms can be broadly categorized into two groups: those based on local feature extraction methods and those centered around extraction methods utilizing interest points. Local feature extraction methods focus on the local features of images, including HOG, Haar, LBP and DPM. A common problem with these methods is that they require the computation of local features for each window, resulting in high computational complexity. On the other hand, the extraction method based on interest points is to extract some feature points or target edges in the image. For example, LoG and Canny algorithms are used to extract edge features, and DoG and Harris methods are used to detect corner features. Despite the speed of these feature extraction techniques, due to the limitations of application scenarios, they are often ineffective in processing images with complex backgrounds or chaotic targets. Therefore, although some traditional object detection algorithms can perform well on certain types of images, their feature robustness is still insufficient when facing targets with diversity and complexity.

2.2 Object detection algorithm based on neural network

The origins of deep learning-powered object detection technology date back to 1998, marked by the creation of a five-layered LeNet network. This network was initially employed for handwritten character recognition, and its pooling layer continues to be prevalent today. Subsequently, in 2012, Hinton introduced AlexNet during the ImageNet Large-scale Image Recognition Challenge, and achieved excellent results through data enhancement, ReLU activation function, and Dropout techniques, marking the widespread interest of convolutional neural networks (CNNS). In recent years, CNN-based object detection methodologies have witnessed rapid advancements, primarily bifurcating into two categories: two-stage detection and single-stage detection.

The basic idea of the two-stage object detection method is to first generate a large number of candidate regions, and then classify and regression each region. In 2014, Girshick et al. proposed RCNN, a

ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

pioneering algorithm that uses selective search methods to obtain candidate regions, AlexNet to extract features, and SVM and multiple regressors to obtain candidate box locations and target categories, with significantly better performance than traditional methods. However, the calculation burden of RCNN is heavy and the detection speed is slow. To this end, He et al. proposed a spatial pyramid pool layer (SPP) to extract features by performing convolution operations on input images, avoiding the operation of feature extraction and normalization for each candidate region one by one. In 2015, Girshick introduced Fast-RCNN based on SPP-NET, combining VGG16 as a convolutional layer and replacing it with an SPP layer to combine classification and border regression and improve detection speed. In 2016, Ren et al. unveiled Faster-RCNN, introducing the Region Proposal Network (RPN) to generate high-quality candidate regions from feature maps, thereby enhancing detection accuracy. Despite continuous improvements in the accuracy of two-stage target detection methods, their detection speed remains a challenge[3-5].

Single-stage object detection is carried out by regression method, which has high detection speed and takes into account the detection accuracy. In recent years, numerous efficient single-stage target detection networks have emerged. In 2014, the VGG-16 network streamlined AlexNet's design by substituting its large pooling and convolutional layers with smaller counterparts, deepening the network's architecture to capture richer feature information. Concurrently, the Google team introduced the Inception network, leveraging a parallel approach to augment network layers and incorporate a copious amount of 1x1 convolutions to diminish feature map dimensions. Additionally, the integration of BN layers for data normalization bolstered the network's convergence capabilities and mitigated overfitting. In 2015, the ResNet network proposed by He Kaiming effectively solved the problem of model performance degradation after deepening the network hierarchy. The residual network realizes identity mapping through short-circuit connection, thus overcoming the problem of gradient disappearance and network degradation. In 2016, the SSD algorithm proposed by Wei Liu learned from the anchor frame mechanism of Faster-RCNN and performed target prediction on the feature graph to generate anchor frame, which improved the detection accuracy. Although the algorithm can effectively retain the target details and semantic information in the upper and lower feature maps, the combination of feature information between different layers is still not ideal, resulting in incomplete feature extraction in small target detection.

In recent years, the detection of small targets, including pedestrians and vehicles, has predominantly relied on deep learning methodologies. In 2016, Redmon introduced the YOLO algorithm, which partitions the input image into S×S grids, with each grid responsible for extracting features and generating multiple candidate bounding boxes, and then determines the target category, location information, and confidence. In 2017, the DenseNet architecture introduced a dense connection module inspired by ResNet, enabling the reuse of each feature layer multiple times, thereby enhancing the network's generalization capabilities. The subsequent year, the creators of YOLO introduced YOLOv3, which substituted DarkNet_19 from YOLOv2 with DarkNet_53 and integrated Feature Pyramid Networks (FPN) to generate multi-scale feature maps, optimizing the detection of small targets. Fast-forwarding to 2020, Bochkovskiy et al. presented YOLOv4, leveraging CSPDarknet53 as the feature extraction network and employing SPP and PANet for feature fusion, effectively reducing the model's parameter count and computational overhead. In the same year, Ultralytics launched YOLOv5, backbone network combined with the Focus module for feature extraction, introduced SPP into the backbone network, and implemented a combined FPN and PAN header network

3. Target tracking algorithm

3.1 YOLO target detection technology

Unlike traditional sliding window and area generation network (RPN) methods, the YOLO algorithm deals with object detection by observing the entire image at once. The core idea is to treat object detection as a regression prediction task, which can directly get bounding boxes and categories from image features. The algorithm initially utilizes a Convolutional Neural Network (CNN) to extract image features, followed by a process of feature fusion. The prediction layer classiifies and regression based on these features to generate the boundary box and class confidence of the target. Specifically, the image to be detected is divided into s×s grid cells, each of which can detect an object whose center point is located in the grid, and generate multiple boundary boxes and their corresponding target class confidence for each grid.

The architecture of YOLOv5 is structured into four core components: Input, Backbone, Neck, and

ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

Output. The Input stage encompasses Mosaic data augmentation, adaptive anchor frame computation, and adaptive image resizing. The Backbone network is constructed with a combination of Focus modules, CBL (Convolutional Layer + Batch Normalization + Leaky ReLU) modules, and SPP (Spatial Pyramid Pooling) modules. The Neck utilizes a hybrid feature fusion approach combining FPN (Feature Pyramid Network) and PAN (Path Aggregation Network). Furthermore, YOLOv5's foundational modules include the CBL module, which integrates a convolutional layer, batch normalization, and a Leaky ReLU activation function; the Resunit residual network module; and the BottleneckCSP module, which incorporates multiple Resunit modules through convolutional layers. Another SPP module realizes multi-scale fusion through maximum pooling. These modules are integrated through tensor concatenation of dimension expansion and tensor addition of dimension preservation, as shown in Figure 1.

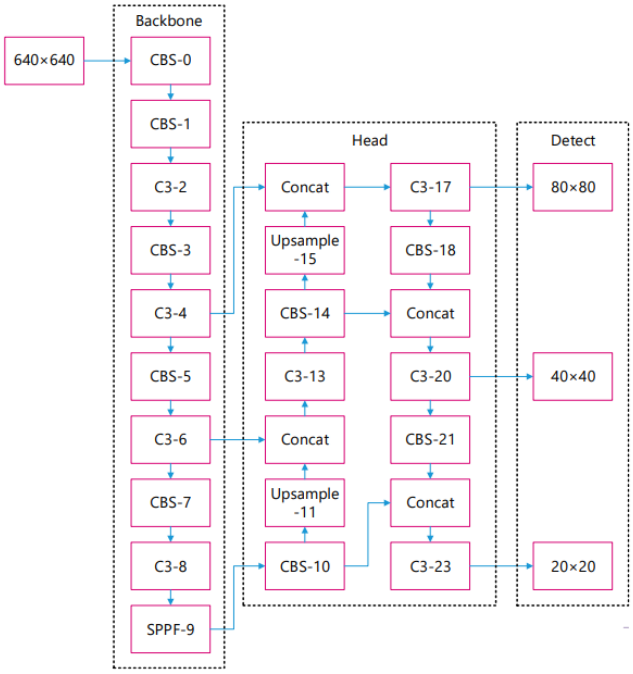

Figure 1: YOLOv5 network structure

3.2 Sort target tracking algorithm

Simple Online and Realtime Tracking algorithm (Sort) is a representative algorithm in target tracking research. Sort algorithm can recognize the same object in adjacent frames by combining the detector and the tracker that predicts the state. This algorithm employs the Kalman Filter to anticipate the target's trajectory, whereas the Hungarian Algorithm achieves the optimal alignment between the detection frame and the tracking frame. Sort algorithm can be updated online, but its performance is highly dependent on the accuracy of the detector. In cases where the target is obscured, similar targets are present, or the movement trend is unclear, Sort's performance may decrease significantly, resulting in target loss or identity confusion[6-7].

(1) Kalman filter

The Kalman Filter is a state estimation technique that incorporates noise and interference into its calculations. The process commences by disregarding noise and interference, fusing a prior estimate of the previous state (i.e., the predicted current state) with observed data (measurements from the detector). This continuous refinement of the prior estimate iteratively leads to a posteriori estimate that more closely approximates the true state. The Kalman filter utilizes a state equation to provide state information and an observation equation to furnish position information. The algorithm consists of two parts: prediction and update, and the optimal result is obtained by iterative estimation.

$$x_w = \left(x, y, r, h, \dot{x}, \dot{y}, \dot{r}, \dot{h}\right)^T \tag{1}$$

$$x_w = Ax_{w-1} + Bu_{w-1} + \omega_{w-1} \tag{2}$$

$$Z_w = (x, y, r, h)^T \tag{3}$$

$$Z_w = Hx_w + V_w \tag{4}$$

ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

Equation 1 represents the vectorial definition of the state x_w within the W frame system, which serves as an estimated value for the system's state. This equation encapsulates the central position of the tracking frame, denoted by (x,y), alongside the aspect ratio (r) and height (h) of the tracking frame. Additionally, it includes four other parameters that correspond to the respective velocity components. In equation of state (3), A, B, u_{w-1} are state transition matrix, control input matrix and system control quantity respectively, andω_{w-1} is Gaussian process noise subject to Q covariance.

Formula (3) is the vector definition of the system observation value Z_w in frame W (the detection value of frame W and the detection frame information of the target), providing the system with position information. In observation equation 4, H signifies the observation matrix, while Vw denotes Gaussian observation noise that is characterized by a covariance matrix R. Figure 2 illustrates the overall progression of the Kalman filtering process.

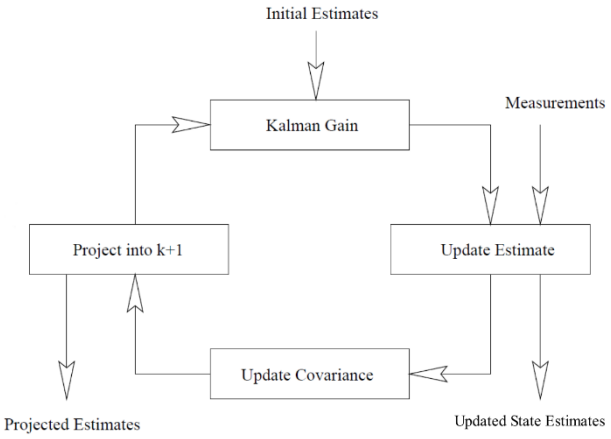

Figure 2: Kalman filter flow

(2) Hungarian algorithm

The Hungarian algorithm, originally proposed by American mathematician Harold W. Kuhn in the 1950s, has found widespread application in solving matching problems within bipartite graphs, particularly in the context of assignment problems. In tracking algorithms, the assignment problem involves identifying the optimal pairing between the current detection responses and the predicted trajectories from the previous frame. The objective is to maximize the number of matches while maintaining a specified level of accuracy. To achieve this, a matching cost matrix C(i, j) is constructed, where each element represents the predicted trajectory i from the previous frame and the current detection result j. The matrix elements are derived from a weighted combination of the motion and appearance features of i and j, and are subsequently quantized to form the metric value d(i, j). Consequently, the problem of matching predictions with detection results is reframed as finding the global optimal solution within the cost matrix, as expressed mathematically in Equation 5. Within the cost matrix, each row and column uniquely features a single smaller element, which signifies the matching cost incurred between a predicted trajectory i and a detection result. This arrangement ensures that the overall cost incurred during the assignment process is minimized.

$$\begin{cases} \min z = \sum_{i=1}^{n} \sum_{j=1}^{n} C_{i,j}\, x_{i,j} \\ \text{s.}\, t. \sum_{i=1}^{n} x_{i,j} = 1, \quad i = 1,2, \ldots, n \\ \sum_{j=1}^{n} x_{i,j} = 1, \mathrm{j} = 1,2, \ldots, n \\ \mathrm{x}_{i,j} = 0 \text{ or } 1, \mathrm{i}, \mathrm{j} = 1,2, \ldots, n \end{cases} \tag{5}$$

In Sort algorithm, the cost matrix is weighted by calculating the IOU (intersection ratio) distance of the objects before and after the two frames, so as to achieve fast calculation. Concurrently, the algorithm incorporates an IOU threshold to assess the validity of matches, thereby filtering out invalid associations. Nevertheless, the Sort algorithm solely relies on the IOU distance for optimally allocating the cost matrix, neglecting the importance of apparent feature matching, which can potentially lead to identity swaps. In scenarios where a target is temporarily occluded or amidst similar-looking targets, the IOU distance may decrease, causing the Kalman filter to either lose track of the target or incorrectly reassign its ID.

ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

3.3 CenterNet network resolution algorithm

CenterNet is an anchor-frameless target detection method based on key point detection. The method does not rely on a prior anchor frame to locate the target, but generates a thermal map on the feature map, identifies the peak points in the thermal map (that is, the target center position), and then generates the target boundary box through the regression of these peak points. In the context of multi-target tracking, CenterNet not only furnishes the central coordinates, width, height, and category of the target detection bounding boxes but also generates the apparent features of the targets. Figure 3 shows the CenterNet network structure adopted in FairMOT, where CBR is the basic network module, including Conv, batch normalization (BN) and ReLU activation function, pooling layer uses maximum pooling to realize downsampling, and main modules include DTB (downsampling tree block) and STB (step tree block). The CenterNet Network structure is shown in figure 3.

CenterNet's backbone feature extraction network uses DLASeg, which integrates FPN structure based on DLA-34 and introduces deformable convolution for up-sampling. This method does not use multi-scale prediction, and only upsamples the feature map 32 times after sampling, and enhances the feature expression ability by combining the features of different layers. Finally, the feature is upsampled to the 1/4 size of the original image before output[8-10].

Because CenterNet does not implement multi-scale prediction, only one branch outputs four messages for the target. This information includes the central location of a class target (heat map), the offset of the central point, the width and height of the target, and the apparent characteristics of the target (Re-ID embedding). The latter provides the necessary embedded information for subsequent target tracking.

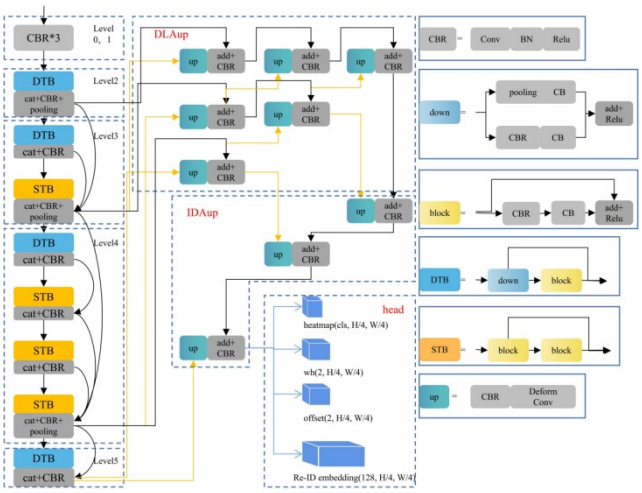

Figure 3: CenterNet Network structure

4. Experiment

4.1 Data set

The dataset utilized in this chapter comprises a segment of the CrowdHuman pedestrian dataset, which was released by Kuangshi. It encompasses 2000 images for training and 400 images for testing purposes. In all pictures, pedestrians are dense and frequently occluded, and the pedestrian scenes are urban streets.

4.2 Experimental setup

The advantages and disadvantages of pedestrian detection algorithms are usually evaluated and quantified by average accuracy (AP), accuracy (Acc) and detection speed (FPS). The accuracy rate (Acc) is calculated as follows:

$$\mathrm{Acc} = \frac{\mathrm{TP+TN}}{\mathrm{TP+FN+FP+FN}} \tag{6}$$

Here, TP stands for the count of positive samples that have been accurately identified, TN signifies

ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

the number of negative samples correctly recognized, FN represents the quantity of positive samples that have been overlooked, and FP denotes the number of negative samples that have been mistakenly classified as positive. CenterNet's backbone network, DLA-34, has been pre-trained on a large pedestrian data set. On this basis, the weight of the backbone network is frozen, and then the DLAup-FEM module is retrained for 30 epochs, followed by fine tuning of the entire network for 140 epochs. The size of the input image is set to 512×512 and the batch_size is set to 12.

4.3 Experimental result

Table 1: Evaluation results on the CrowdHuman dataset

Model	$AP_{0.5}$/%	Acc/%
Sort	79.64	77.52
YoLo+Sort	84.31	83.82
CenterNet	85.57	84.16

Table I assesses the performance of different models by two key metrics: Average Precision (AP0.5) and Accuracy (Acc). Specifically, we focus on how the SORT model, the YOLO+SORT combination model, and the CenterNet model perform on these two metrics. The AP0.5 of SORT model is 79.64% and the accuracy is 77.52%. This shows that SORT has high average accuracy and accuracy when dealing with target tracking tasks, but there is still room for improvement. The model combining YOLO target detector and SORT tracker achieves 84.31% on AP0.5, and the accuracy is 83.82%. Compared with the pure SORT model, the YOLO+SORT combined model has significant improvement in both indexes, indicating that the addition of YOLO detector effectively improves the tracking performance.

The CenterNet model achieved 85.57% on AP0.5 with an accuracy of 84.16%. Compared with the other two models, CenterNet has the best performance on both indicators, which indicates that the model has higher accuracy and robustness in object detection and tracking. It can be seen from the above data that with the increase of model complexity, the performance of target tracking also improves. Therefore, in practical applications, if the pursuit of higher accuracy and average accuracy, CenterNet is a better choice.

Figure 4: CenterNet tracking results with occlusions

Figure 4 shows the target tracking results with occlusions. From the results, it is clear that the CenterNet algorithm is able to detect the target very well even though the target is in a moving state and will pass through the obscured area. This strongly proves the effectiveness and robustness of CenterNet algorithm in processing occluded data. Even in complex scenes, the algorithm can maintain stable performance and ensure that the target will not lose track because of short occlusion, which is crucial for the target tracking task in practical applications[11-12].

5. Conclusion

This paper reviews the research progress and application of target tracking algorithms based on computer vision. Firstly, we introduce the basic concept and technical background of target tracking, including the key technical links of target detection, feature extraction and motion prediction. Then, we analyze several mainstream target tracking methods in detail. This paper emphasizes the application of deep learning technology in the field of target tracking, and introduces some advanced tracking

ISSN 2616-5775 Vol. 7, Issue 9: 74-81, DOI: 10.25236/AJCIS.2024.070910

algorithms based on neural network. The effectiveness of these algorithms in dealing with challenges such as occlusion, illumination variation and scale variation is verified by experiments.

References

[1] Konstantinova, Pavlina D., Alexander Udvarev, and Tzvetan Semerdjiev. "A study of a target tracking algorithm using global nearest neighbor approach." Compsystech. Vol. 3. 2003.
[2] Uhlmann, Jeffrey K. "Algorithms for multiple-target tracking." American Scientist 80.2 (1992): 128-141.
[3] Souza, Éfren L., Eduardo F. Nakamura, and Richard W. Pazzi. "Target tracking for sensor networks: A survey." ACM Computing Surveys (CSUR) 49.2 (2016): 1-31.
[4] Fiaz, Mustansar, Arif Mahmood, and Soon Ki Jung. "Tracking noisy targets: A review of recent object tracking approaches." arxiv preprint arxiv:1802.03098 (2018).
[5] Zefenfen, et al. "Target tracking approach via quantum genetic algorithm." IET Computer Vision 12.3 (2018): 241-251.
[6] Kim, JeongWoon, and David Hyunchul Shim. "A vision-based target tracking control system of a quadrotor by using a tablet computer." 2013 international conference on unmanned aircraft systems (icuas). IEEE, 2013.
[7] Kamkar, Shiva, et al. "Multiple-target tracking in human and machine vision." PLoS computational biology 16.4 (2020): e1007698.
[8] Jeong, Jaehoon, et al. "Mcta: Target tracking algorithm based on minimal contour in wireless sensor networks." IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications. IEEE, 2007.
[9] Arnold, James, S. W. Shaw, and H. E. N. R. I. Pasternack. "Efficient target tracking using dynamic programming." IEEE transactions on Aerospace and Electronic Systems 29.1 (1993): 44-56.
[10] Rameshbabu, K., et al. "Target tracking system using kalman filter." International Journal of Advanced Engineering Research and Studies 2 (2012): 90-94.
[11] Fan, Litong, et al. "A survey on multiple object tracking algorithm." 2016 IEEE International Conference on Information and Automation (ICIA). IEEE, 2016.
[12] Khedr, Ahmed M., and Walid Osamy. "Effective target tracking mechanism in a self-organizing wireless sensor network." Journal of Parallel and Distributed Computing 71.10 (2011): 1318-1326.