IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received August 31, 2020, accepted September 10, 2020, date of publication September 18, 2020,
date of current version December 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3024928

Multi-Target Tracking and Detection Based on Hybrid Filter Algorithm

XIANZHEN XU[1], ZHIYU YUAN[2], AND YANPING WANG[3]

[1]Laboratory of Fiber Materials and Modern Textile, Instrumental Analysis Center, Shandong Sino-Japanese Center for Collaborative Research of Carbon Nanomaterials, Collaborative Innovation Center for Marine Biomass Fiber Materials and Textiles, College of Chemistry and Chemical Engineering, Qingdao University, Qingdao 266071, China
[2]College of Engineering, Michigan State University, East Lansing, MI 48823, USA
[3]China Tobacco Zhejiang Industry Company Ltd., Hangzhou 310008, China

Corresponding author: Zhiyu Yuan (yuanzhiy@msu.edu)

ABSTRACT Wireless sensor network is a fast-growing research field. In recent years, it has attracted considerable research attention. The generation of large-scale sensor networks interconnected hundreds of sensor nodes, opening up some technical challenges and huge application potential. This article mainly introduces multi-target tracking and detection based on wireless sensor network. This paper studies the positioning methods and target tracking algorithms of wireless sensor network nodes. It mainly studies the positioning algorithms based on ranging in the positioning algorithms in detail, analyzes the advantages and disadvantages of the algorithms, and analyzes the system models and related targets in the target tracking algorithms. The filtering has been studied in detail. In addition, a tracking algorithm under the mixed linear and non-linear motion of the moving target is also proposed, namely the hybrid filtering algorithm. This algorithm makes the motion state of the tracked moving target no longer restricted, and can freely switch between linear motion and nonlinear motion. The experimental results in this paper show that Kalman filter can effectively track moving targets without sudden changes in speed. When the mobile robot switches the grid, it will bring about the switching of the observation model. Compared with the least squares positioning algorithm, the smooth switching rate of the Kalman filter positioning algorithm is increased by 24%. When the three robots are running at a speed of 0. 5m/s in the monitoring area, the system can track the target in real time and send the positioning result to the robot to provide position navigation for the next formation feedback control.

INDEX TERMS Wireless sensor network, multi-target tracking, target tracking algorithm, target detection GMM algorithm.

I. INTRODUCTION

With the rapid development and maturity of hardware manufacturing and software development technologies, such as embedded technology, micro-electromechanical systems, wireless communication technology, and sensor technology, the development and application of highly integrated, multi-functional and miniaturized sensor nodes has become may. These sensor nodes have the capabilities of information collection, data processing and wireless communication. They can form a network through free organization and connection, and cooperate to sense, collect, process and transmit various sensing data. This self-organizing network is called a wireless sensor network.

At the beginning of the transition from military to civilian use, wireless sensor networks were mainly used in environmental monitoring. With the development of research, wireless sensor networks have obvious advantages in target tracking applications. The densely deployed sensor nodes can accurately sense, track and control the moving target, so that the movement of the moving target can be displayed in more detail. Due to the autonomy, self-organization, and high-density deployment of wireless sensor networks, when nodes fail or new nodes join, they can be automatically deployed and fault-tolerant in harsh environments, making wireless sensor networks more reliable when tracking targets. Fault tolerance and robustness. Synchronous monitoring of multiple sensors

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

makes the discovery of moving targets more timely and easier. Distributed data processing and coordinated work of multiple sensor nodes make tracking more comprehensive.

The wireless sensor network(WSN) proposed by Hammoudeh M is a low-cost technology that provides a smart-led solution that can effectively continuously monitor large, busy, and complex landscapes. The linear network topology created by the structure of the surveillance area presents challenges that have not been fully resolved in the literature to date. In addition, he also determined an appropriate metric to measure the quality of WSN boundary detection. However, the complexity of wireless sensor networks is too high, making the whole process very complicated [1]. The unique characteristics of the acoustic communication channel proposed by Farwa A (such as high propagation delay, multipath fading, rapid attenuation of acoustic signals, etc.) limit the use of underwater wireless sensor networks (UWSN). However, the immutable choice of the transponder node will lead to the sudden death of the node, resulting in an imbalance of energy consumption and voids [2]. Soderlund A A proposed a new fast and efficient clustering algorithm for sensor nodes in heterogeneous wireless sensor networks. The goal is to make the optimal configuration of the sensor reduce the positioning uncertainty in multi-target tracking. The algorithm is based on three indicators: 1) perceived feasibility, 2) measuring quality to maximize information utility, and 3) communication cost to minimize data routing time. The derived cluster is used as the search space of the sensor. By optimizing the differential entropy, the uncertainty of the target prediction posterior distribution is effectively reduced. But this method is not very accurate, and there will be errors in the results [3].

The innovation of this paper is (1) combines the geometric characteristics of the binary sensor network model and the error restriction conditions, improves the quality of sampled particles, filters out redundant particles, and improves the accuracy of target tracking. (2) Summarize the current domestic and foreign related issues of wireless sensor network target detection and tracking protocol algorithms, and analyze the advantages and disadvantages of various protocol algorithms for different target detection problems.

II. MULTI-TARGET TRACKING AND DETECTION ALGORITHM

A. TARGET TRACKING ALGORITHM

Tracking targets requires the use of state-space methods to model dynamic systems. To describe a dynamic system, at least two models are needed: one is the model that describes the state transition, that is, the system model; the other is the state measurement equation [4]. For the Bayesian method of dynamic estimation, it is necessary to use all available information to construct the posterior density function of the state [5]. If the system model or measurement model is nonlinear, then the posterior density function will be non-Gaussian [6]. In principle, the best estimate of the state may be included in the posterior density function. For many problems, an estimate is required for every measurement [7]. An iterative filtering method means that the received data is obtained sequentially rather than all at once, so it is not necessary to store a complete data set [8]. Such a filter consists of two important stages: prediction and update. The prediction stage uses the system model to predict the state density function of the next stage [9]. The update phase uses the latest measured value to modify the predicted density function. This is achieved by the Bayesian theory of a mechanism for updating the target state based on additional information from new data [10].

For nonlinear filtering, the target state vector A_x is introduced, where A_n is the dimension of the state vector, f is the set of real numbers; x is the time index; N is the set of natural numbers. The target state is expressed by the following discrete event model:

$$A_x = f_{x-1}(A_{x-1}, V_{x-1}) \tag{1}$$

Among them, f_{x-1} is the determined non-linear function of state a_{x-1}, process noise is represented by V_{x-1}, and the target measurement equation is:

$$z_x = h_x(A_x, W_k) \tag{2}$$

h_x is a known nonlinear equation, W_x is the measurement noise [11]. The noise sequences V_{x-1} and W_x are assumed to be Gaussian white noise, and the probability density equation is known and independent of each other. Assume that the initial target state has a known initial probability density function $P(y_0)$, which is independent of noise [12].

We filter and estimate A_x based on all available measurements. From the perspective of Bayesian theory, this problem is to iteratively obtain a certain degree of credibility under a given situation. Therefore, this requires the construction of the posterior probability density function [13]. The initial density function of the state vector, where z_0 has no measured value. Then, the probability density function $P(A_x|Z_x)$ can be obtained from the two steps of prediction and update mentioned earlier [14].

Assuming that the required probability density function is available, for the system model (1) the predicted density function is obtained through the C-K equation at time x:

$$P(A_{x-1}|Z_{x-1}) = \int P(A_x|Z_{x-1})P(A_{x-1}|Z_{x-1})dA_{x-1} \tag{3}$$

When the measured value is available at time x, then the prediction step can be implemented, which involves an update under the prediction probability density function under Bayes' rule:

$$\begin{aligned} P(A_x|Z_x) &= P(A_x|Z_x, Z_{x-1}) \\ &= P(A_x|Z_x, Z_{x-1})P(A_x|Z_{x-1})P(z_x|Z_x) \\ &= P(z_x|A_x)P(A_x|Z_{x-1})P(z_x|Z_{x-1}) \end{aligned} \tag{4}$$

Regularization constants are available:

$$P(A_x|Z_x) = \int P(A_x|Z_x)P(A_x|Z_{x-1})dA_x \tag{5}$$

B. TARGET DETECTION GMM ALGORITHM

In order to adapt to the above-mentioned environment, some people proposed the GMM algorithm [15]. The idea of the GMM algorithm is: set K different situations to describe the color content of the same pixel in a video image sequence [16]. The higher the K value, the more models, the stronger the anti-interference ability, and the more stable the background modeling result. However, as the K value increases, the complexity of the algorithm will increase and the speed of background modeling will also slow down, so the K value generally lasts 3-5 [17]. The probability density function of the GMM algorithm is expressed as:

$$g(f_t) = \sum_{i=1}^{k} w_{i,t} * \eta_i(f_t, \bar{f}_{i,t}, \sum{}_{i,t}) \tag{6}$$

In formula (1), f_t represents the color value of the pixel at the current position at the current moment, $w_{i,t}$ and $\eta(f_t, \bar{f}_{i,t}, \sum_{i,t})$ respectively represent the weight and probability density function of the i-th Gaussian model of the current pixel, and $w_{i,t}$ and $\sum_{i,t}$ are respectively the i-th Gaussian density function when Expected value and variance at the previous moment [18]. In order to meet the requirements of the probability density function, the sum of the weights of the K Gaussian models needs to be 1, namely:

$$\sum_{i=1}^{K} w_{i,t} = 1 \tag{7}$$

In the GMM algorithm, each Gaussian probability density function component can be described by the following formula:

$$\eta_i(f_t, \bar{f}_i, \sum{}_{i,t}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\sum_{i,t}|^{\frac{1}{2}}} e^{-\frac{1}{2}(f_t, -\bar{f}_i) \sum_{i,t}^{-1} (f_t, \bar{f}_i)} \tag{8}$$

In formula (3), n represents the dimension of the image color, and the variance matrix is:

$$\sum{}_{i,t} = a_{i,t}^2 I \tag{9}$$

$a_{i,t}$ is the standard deviation of the i-th Gaussian model at the current moment [19]. In the RGB color image obtained in this paper, R = G = B, so it is directly converted to a gray image, and a Gaussian model is established for it, so n = 1, the variance and standard deviation a are scalars. Can be reduced to:

$$\eta_i(f_t, \bar{f}_i, \sum{}_{i,t}) = e^{-\frac{1}{2}(f_t, -\bar{f}_i) \sum_{i,t}^{-1} (f_t, \bar{f}_i)} \frac{1}{a_{i,t}\sqrt{2\pi}} e^{-\frac{(f_t, -\bar{f}_i)^2}{2a_{i,t}^2}} \tag{10}$$

When initializing the background model, the average gray level in the video sequence images and the change of each image over a period of time are usually used to initialize the K states of the Gaussian distribution from each pixel [20].

The update of the Gaussian mixture model is mainly the Gaussian parameter value of the Gaussian distribution function of each pixel in the background model, including weight, average, average, etc., parameter variability and learning rate [21]. The update of model parameters is related to whether the current pixel matches the Gaussian model in the background. As:

$$|f_t - \bar{f}_{i,t-1}| < C_G a_{i,t-1} \tag{11}$$

Indicates that the pixel point matches the i-th Gaussian model, then this point is a background pixel, record the matching position, and update the parameters of the Gaussian model at the corresponding position:

$$\bar{f}_{i,t} = (1 - p_{i,t}) + \bar{f}_{i,t-1} + p_{i,t} f_t \tag{12}$$

$$a_{i,t}^2 = (1 - p_{i,t}) a_{i,t-1}^2 + p_{i,t}(f_t - \bar{f}_{i,t})^2 \tag{13}$$

In formula (9), $M_{i,t}$ is a binary number. If the current pixel matches any Gaussian model at the current position, its value is 1; otherwise, its value is 0. a represents the model learning rate, which needs to be customized by the user [22]. Generally, the value is 0, 1, and its size is related to the background model update speed. $\frac{1}{a}$ represents the fixed time of the algorithm update interval, and $p_{i,t}$ represents the parameter learning rate of the i-th Gaussian model. In other Gaussian models, except the weight decreases, other parameters remain unchanged:

$$w_{i,t} = (1 - a) w_{i,t-1} a M_{i,t} \tag{14}$$

If the current pixel value does not match the K Gaussian model, replace the Gaussian model with the smallest weight with the new Gaussian model, and give the new Gaussian model a lower value [23]. Because both weight and change can indicate whether image pixels are background pixels, according to the selection method of the update rate, the distribution is classified in descending order.

C. TARGET LOCATION ALGORITHM OF WIRELESS SENSOR NETWORK

1) TARGET POSITIONING TECHNOLOGY BASED ON RSSI

When the radio wave signal propagates in the atmospheric space, there will be different degrees of loss. The RSSI ranging technology uses the attenuation of the signal in the propagation process to estimate the distance between nodes. The RSSI signal power receiver can receive specific signal power in space, and the received signal power is a function of time, frequency and space. Signal loss is generally divided into path loss, shadow slow fading and multipath fast fading [24]. The path loss is mainly the propagation loss caused by the distance between the signal power transmitter and the receiver, and the slow shadow fading is the change in the average signal strength caused by the complex and changeable terrain environment. Multipath fast fading is related to the signal strength characteristics of signals closer to the receiver.

RSSI-based positioning technology mainly measures the difference between the signal strength transmitted by the transmitter and the signal strength received by the receiver, and then converts the change in signal strength into the distance between the target and the receiver according to the channel attenuation model. When radio wave signals propagate in free space, a shadowing model is generally used to characterize changes in signal strength [25].

2) TARGET POSITIONING TECHNOLOGY BASED ON TDOA

TDOA positioning technology is to simultaneously transmit two types of radio wave signals with different speeds from the transmitting end, and estimate the distance between the transmitting end and the receiving end based on the time difference between the two types of signals reaching the receiving end. The radio wave signals commonly used in TDOA-based ranging and positioning algorithms are radio frequency signals and ultrasonic signals. It should be pointed out that compared with TOA ranging technology, TDOA ranging technology does not require time synchronization between the transmitter and the receiver [26].

III. WIRELESS SENSOR MULTI-TARGET TRACKING EXPERIMENT

A. WIRELESS SENSOR NETWORK EXPERIMENT

Comparing the algorithm with Matlab2014a platform using Mica2 sensor node, material 2 = 0. 0125, PT = 0dB, d0 = 1, 0m, PL(d0) = 50dB, n = 4. A random node network is established in the 100-x100m monitoring area, and the model is used to establish a grid scanning algorithm to detect errors and an improved grid scanning algorithm is broadcast through a true edge distribution model and a BP neural network. In the experiment, the traditional grid scanning algorithm is referred to as GS, and the improved grid scanning algorithm in this paper is referred to as TGS. The average correlation detection error normalized by unknown nodes in the computer network should be used .

Where: (xi, yi) represents the position coordinates of unknown nodes, (xi, yi) represents the actual coordinates of unknown nodes, r represents the communication radius of the nodes in the network, Nu represents the number of unknown nodes that can be detected in the network, k Represents the rules and regulations of the number of nodes randomly created in the network. In the experiment, the number of normalized modes k = 100 means that 100 random node developments are generated. Allow the number of nodes in the network to be N, the number of nodes to be A, and the length of the grid edge to be.

In the experiment, each time the network topology is created randomly, the Gaussian random XS variable caused by the weakening of the shadow is different, the loss coordinator is also different, and the path Ki is different for each Mica2 node. Therefore, the XS and Ki parameters generated each time are stored in the experiment, and the detection errors of the GS and TGS algorithms before and after installation are simulated; the BP neural network improves the credibility of the experiment.

B. TARGET TRACKING EXPERIMENT OF WIRELESS SENSOR NETWORK

This experiment uses Matlab to carry out the simulation experiment, and what needs to be considered is the target tracking based on the sensing distance algorithm under ideal conditions. Due to its low algorithm complexity, this article adds routing errors and attacks on the basis of this algorithm, discusses the ideal sensing distance algorithm (algorithm a), non-ideal sensing distance algorithm without detection mechanism (algorithm b), and non-ideal addition The performance of the sensing distance algorithm (algorithm c) for information detection. The simulation environment is: 500 sensor nodes are scattered in the $10 \times 10m^2$ monitoring area, and the sensor node's sensing radius is 10 m, and the communication radius is 5 m. The initial position of the target is (20, 100). At the beginning, the initial speed is 0m/s in the direction of the x axis, and the linear motion is performed on the y axis at a speed of 3 m/s, then the uniform acceleration motion is performed, and finally the uniform velocity is performed movement.

Since the sampling frequency, target speed and network status are the factors that affect the disorder, the simulation first uses the target to move at a constant speed to keep the target speed constant to study the influence of different sampling frequencies on the error. As shown in Figure 4, in a certain period of time, too many sampling times increase the disorder rate, but too few acquisition times affect the positioning accuracy, so the appropriate sampling times should be selected. This article takes 500 sampling times.

C. PRECISE POSITIONING OF WSN CLUSTERING MODEL

A wireless sensor network (WSN) is an autonomous network composed of a large number of small, low-cost sensor nodes that integrate sensing, data collection, processing, and wireless communication capabilities. A typical hierarchical WSN is composed of sensor nodes (Sensor Node, SN), cluster head nodes (Cluster Head, CH) and base stations (Base Station, BS).

The classic IDSQ algorithm is based on the fact that when the target moves in the geographic area covered by the wireless sensor network, its trajectory is continuous in time and space. Once the target enters the geographic area, it is already in the monitoring range of a certain sensor node; when the target leaves the monitoring range of a node, it immediately enters the monitoring range of its neighboring nodes until the target leaves the geographic area covered by the sensor network. Let each node record the status and monitoring time of the monitored target. In this way, the sensor network records the trajectory of each target. Due to the limited storage space of sensor nodes, if too many targets have been monitored, the information of all targets may not be saved, and due to energy problems, sensing range, estimation capabilities, bandwidth and energy constraints, each sensor node cannot be the whole process Record information about the status of the target. In this regard, we divide the sensor nodes into different clusters according to the location and energy of the nodes in the wireless sensor network. Each cluster is composed of a cluster head node and a number of ordinary nodes. The ordinary nodes in the cluster only report to the cluster head to which they belong. The node transmits data, and the cluster head node transmits data to the sink node level. From a logical point of view, the nodes that were originally equal are divided into different levels. The cluster head node with higher energy or closest to the target position is at a higher level and undertakes more tasks, not only for perception and upper level The node transmits its own new data, and also transfers the data passed by the lower node, and performs data fusion. The ordinary node with lower energy and far away from the target location is at a lower level, and only performs the task of sensing and transmitting to the upper node.

Assuming that the sensor nodes are clustered in the form of virtual cells, each virtual cell in the clustering model has a cluster head node CH (Cluster Head). The cluster head node can receive messages sent by nodes in the cluster, and Fusion, sometimes only need to ensure that a cluster has only one active node CH, and other nodes are in a sleep state and take turns as cluster head nodes. This can greatly save the energy consumed by the entire network and extend the life cycle of the entire sensor network. The cluster model is for ease of discussion.

D. TARGET TRACKING AND POSITIONING EXPERIMENT

The experiment will verify the effectiveness of the proposed CS-based SAOMP algorithm through experiments in actual positioning and monitoring scenarios. The monitoring area is defined as a square area 63, which targets three people of different heights. Eight sensor nodes are evenly developed on both sides of the monitoring area. The radio communication unit used is the radio communication unit CC2530. One of the sensor nodes is used as a cluster node, which is mainly responsible for transmitting the collected data to the computer. Run data processing.

In the experiment, the heights of the three targets were 1 meter, 1 meter, 3 meters, and 1. 6 meters, the height of the sensor node was 0. 8 meters, and the actual heights of the three targets relative to the sensor launch node were 0, 2 m, 0, and 5 m, respectively. And 0, 8m. These three goals are called Goal 1, Goal 2, and Goal 3 in the series. CS-based SAOMP passive location algorithm mainly includes two steps:

The first step is to build the perception matrix. First, before the target does not enter the monitoring area, let the i-th link record the RSSI measurement value, and use F_i to represent the average value of the recorded RSSI measurement value. Then, let target 2 traverse all N grids in the monitoring area. When the target is the j-th grid, let the i-th link record the RSSI measurement value, and use R_{ij} to represent the recorded RSSI measurement value average of.

The second step is to set goals. Objective To randomly assign K (3 persons, dilution K) to K grids in the monitoring area. In order to make the experiment closer to the actual layout scene, unlike the previous layout algorithm search, the target is set at the center of the grid. Let the target stand randomly anywhere on the grid. When the target exists, the MSI connection is used to record the RSSI measurement value to form the measurement vector $Y_{M\times 1}$, and transfer all the collected data to the computer through the convergence node, and restore the target position vector Q through the SAOMP algorithm to achieve the positioning of the target.

TABLE 1. Setting of simulation parameters.

Parameter name	Value
Sampling period T	0. 1 s
Shortest delay t	10 m s
Max of random delay (tran)	1 ms
Select cluster head time threshold TTH	50 m s
Cluster head node built-in threshold n	25

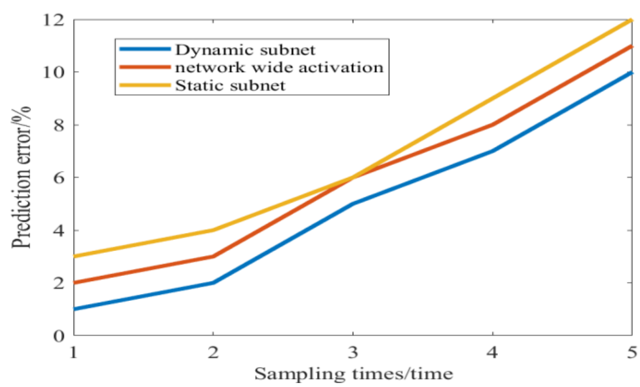

FIGURE 1. Estimated error of the three strategies.

IV. SENSOR MULTI-TARGET TRACKING ANALYSIS

A. ANALYSIS OF SENSOR NETWORK RESULTS

Through statistical analysis, the network data can be obtained as follows: In a 50 × 50 × 50 area, 1000 sensor nodes are evenly arranged, the sensing radius between the nodes is 5, and the communication radius is 10. At the same time, suppose that a target with an energy of 1000 moves in a straight line at a speed of 6m per second, enters from the point (4, 60, 50), and leaves the area from the point (30, 2, 20). Other parameters are shown in Table 1.

The error of target position estimation under three strategies is shown in Figure 1.

The figure above shows that the projected dynamic infrastructure algorithm has a relatively small estimation error. This is because all activation algorithms will cause more values to deviate from the goal of participating in the position evaluation, resulting in greater errors. The scale and scale of the fixed network algorithm complex should be stable. When the target moves to the edge of the infrastructure, due to incorrect header settings and other reasons, the target enters the regional infrastructure and detects that the node is inactive and cannot complete network monitoring. The dynamic infrastructure strategy can dynamically adjust the

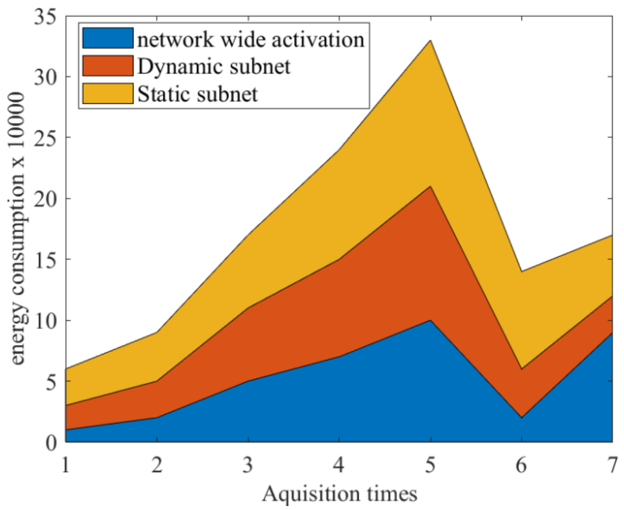

FIGURE 2. Energy consumption of the three policies.

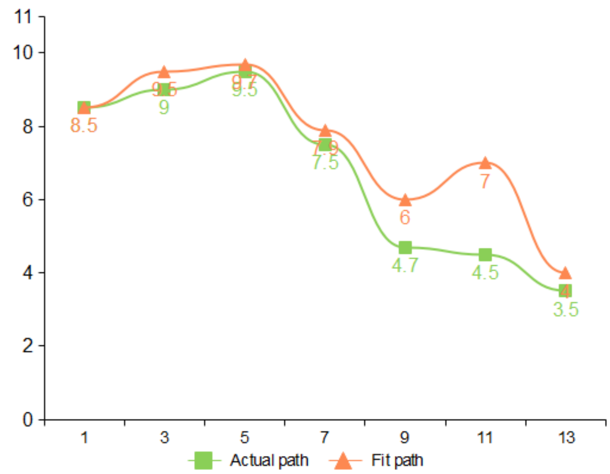

FIGURE 3. Effect picture of path selection.

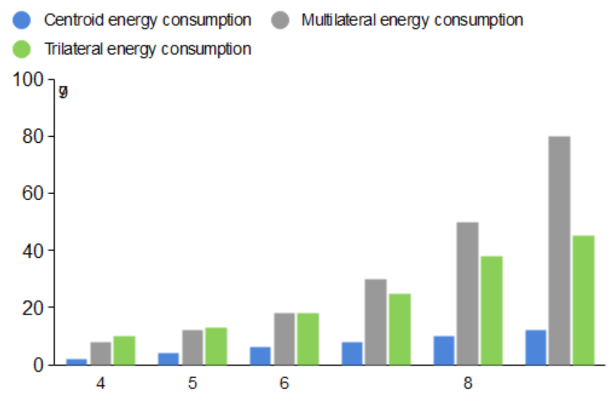

FIGURE 4. Target positioning energy consumption comparison.

TABLE 2. Comparison of positioning errors of three algorithms.

Number of groups	A1	A2	A3	A4
centroid	0. 55	0. 75	0. 66	0. 38
Trilateral	0. 42	0. 34	0. 37	0. 4
multilateral	0, 16	0. 12	1. 14	0. 15

complex's network according to the location of the target for target detection, detection, and monitoring .

The energy consumption under the three strategies is shown in Figure 2. Figure 2 shows that when monitoring targets, the algorithm has lower energy consumption than the other two strategies. Both this algorithm and the fixed infrastructure algorithm must activate candidate nodes in the target area and select the head node to be complex. Assuming that the two algorithms consume the same energy in this process, the energy consumption of the stable infrastructure algorithm is even higher than that consumed by the entire network. In addition, although the strategy of activating the entire network does not require this part of the loss, in target detection, the energy consumed by the entire network activation is much higher than the energy consumption of this part. Therefore, in order to reduce the energy consumption of the network, the algorithm is designed to reduce the number of activated nodes as much as possible while ensuring low error.

B. ANALYSIS OF MULTI-TARGET TRACKING RESULTS

Through analysis, the error comparison of the three algorithms for positioning can be obtained, as shown in Table 2:

The analysis and comparison in Table 2 show that the position of the center of mass is very different from the positions of the three views. Due to the interference of many uncertain factors, the layout results are unreliable. The algorithm and multilateral positioning algorithm proposed in this paper have the advantages of narrow positioning range and high positioning accuracy. This is due to the coordination of multiple nodes and the two-step optimization method, which weakens the impact of large error nodes on accuracy placement.

On the basis of the coordinate information at each time obtained by the positioning algorithm, the running track of the target can be obtained by curve fitting.

Figure 3 is a roadmap for target monitoring following the model selection mechanism described above. From this shape, it may solve the problem of path distortion caused by quadratic curve target tracking. If a curve with a level greater than 2 is used to adjust the route, even if the positioning accuracy is very high, it is still necessary to reduce expenses when dealing with complex or long journeys. The result is similar to the second-order segmented layout curve, but the energy consumption is much higher than the second-order curve.

From the simulation results shown in Figure 4, it can be seen that the energy loss of the two-step target placement algorithm also increases with the position node and width, but compared with the polymer placement algorithm, in general, the sensor network saves a lot of energy . A comprehensive comparison of the four layout algorithms. The calculations of these four algorithms are not complicated and are suitable for execution on sensor network nodes. The center and three-sided layout algorithm saves more energy, but the layout accuracy is lower. power consumption. The two-step optimization algorithm proposed in this paper can also obtain relatively ideal positioning accuracy, and consider the energy cost, which has high practical value.

C. ANALYSIS OF TARGET LOCATION ALGORITHM

From the statistical data, the evaluation results of the OMP, SAMP and SAOMP algorithms on the target position can be obtained. From the position evaluation results of the three targets, the position evaluation error of target 2 is the smallest, and the position evaluation errors of target 1 and target 3 are the smallest. The bigger reason is that it is mainly due to the early development stage, and goal 2 spans all networks in the monitoring area. For the construction of Table A, the effective heights of the three targets are different from the sensor nodes. Therefore, the actual size of the target is one of the factors that affect the accuracy of target setting. Studying the effect of the actual height of the target on the positioning accuracy will be our next step. jobs. In general, the OMP algorithm has the worst performance when extracting the target Q position vector. The main reason is that the OMP algorithm must first obtain the dilution Q in the process of recovering the dilution Q, and the dilution is unknown in the unit. The proposed SAOMP algorithm has the best performance.

Therefore, two targets with different motion directions and trajectories can also be monitored. Let target 2 pass through the four sides on the left side of the rectangle at a uniform speed, and let target 3 turn to the right at a uniform speed. First, use the SAOMP algorithm to estimate the target position within time t; then, use the OCKF algorithm to predict and estimate the target's next position, and monitor the target throughout the process; observe. It can be seen from the result data that the target monitoring format using the combination of SAOMP algorithm and OCKF algorithm is feasible, basically meets the default requirements, and the monitoring error is small. Since the experiment is conducted in an open external environment, some random factors will cause large monitoring errors in a short time. In the future, we will focus on how to improve the robustness of the algorithm. Similarly, the accuracy of monitoring target 2 is also higher than that of target 1. The reason is that target 2 constructs the perception matrix A across all networks in the surveillance area.

The data can then be used to show the effect of the target number (K axis) on the placement result. The detection error is used to evaluate the performance of different recovery algorithms for the target location. This paper compares the positioning performance of OMP, SAMP and SAOMP algorithms. The results show that as the number of targets (dilution K) increases, the detection error gradually increases, especially the performance of the OMP algorithm is more obvious. The reason for the poor performance of the OMP algorithm when identifying multiple targets is that you should know the vector dilution K in advance, ie. The target number and target number of carrier signal search in this module are unknown. Both the SAMP algorithm and the SAOMP algorithm can estimate the dilution of the cross section, so the positioning performance is similar. However, the SAMP algorithm must gradually approach the initial signal with a fixed step size. How to choose the step size is a difficult problem to solve .

V. CONCLUSION

This article mainly introduces the overall architecture of the system platform, related material elements, and the software involved, such as the OpenCV function library, classifies the project implementation process blocks and program flowcharts. Then it is divided into three parts according to the functions of target detection and monitoring: target detection, target positioning and target monitoring, which are described in detail. In this paper, the theoretical basis, experimental results and comparative description of the algorithms on the functional platform of each functional unit are given in detail.

This article introduces the situation of introducing multiple similar moving targets into the monitoring network at the same time, and analyzes the possible problems and causes. At the same time, the multi-target signal generated from the network node is collected to calculate a signal alias. Based on the potential personal target monitoring system, according to actual needs, multiple dynamic monitoring sets are combined into one or more dynamic monitoring sets. Aiming at the problem of very random target behavior when multiple targets are monitored centrally, it is impossible to use the path estimation process of a single target monitoring process and a circle of "buffer" around the dynamic complex; for dynamic maintenance of dynamic assembly projects.

This article introduces several important classic state evaluation algorithms, such as small block evaluation algorithm, Kalman filter estimation algorithm, a new state evaluation algorithm ECF filter estimation and particle filter algorithm. Particulate filtration technology is suitable for target state assessment under non-linear and non-standard conditions, and thus has been widely used. However, particle filtering also has the problems of large amount of calculation, particle degradation and particle exhaustion during particle monitoring. And improved the sampling process in the particle filtering algorithm. The sampling algorithm can not only suppress the degradation of particulate matter during the monitoring process, but also effectively reduce the calculation of particulate matter filtration.

REFERENCES

[1] S. Wan, Z. Gu, and Q. Ni, "Cognitive computing and wireless communications on the edge for healthcare service robots," *Comput. Commun.*, vol. 149, pp. 99–106, Jan. 2020.

[2] W. Elsayed, M. Elhoseny, S. Sabbeh, and A. Riad, "Self-maintenance model for wireless sensor networks," *Comput. Electr. Eng.*, vol. 70, pp. 799–812, Aug. 2018.

[3] M. Pan, Y. Liu, J. Cao, Y. Li, C. Li, and C.-H. Chen, "Visual recognition based on deep learning for navigation mark classification," *IEEE Access*, vol. 8, pp. 32767–32775, 2020.

[4] M. Elhoseny, "Multi-object detection and tracking (MODT) machine learning model for real-time video surveillance systems," *Circuits, Syst., Signal Process.*, vol. 39, no. 2, pp. 611–630, Aug. 2019.

[5] M. Hammoudeh, F. Al-Fayez, and H. Lloyd, "A wireless sensor network border monitoring system: Deployment issues and routing protocols," *IEEE Sensors J.*, vol. 17, no. 8, pp. 2572–2582, Aug. 2017.

[6] S. N. Mohanty, E. L. Lydia, M. Elhoseny, M. M. G. Al Otaibi, and K. Shankar, "Deep learning with LSTM based distributed data mining model for energy efficient wireless sensor networks," *Phys. Commun.*, vol. 40, Jun. 2020, Art. no. 101097.

[7] F. Ahmed, Z. Wadud, N. Javaid, N. Alrajeh, M. Alabed, and U. Qasim, "Mobile sinks assisted geographic and opportunistic routing based interference avoidance for underwater wireless sensor network," *Sensors*, vol. 18, no. 4, p. 1062, Apr. 2018.

[8] A. A. Soderlund and M. Kumar, "Optimization of multitarget tracking within a sensor network via information-guided clustering," *J. Guid., Control, Dyn.*, vol. 42, no. 2, pp. 317–334, Feb. 2019.

[9] H. Ali, "Energy efficient hierarchical clustering mechanism for wireless sensor network fields," *Int. J. Comput. Appl.*, vol. 153, no. 8, pp. 42–46, Nov. 2016.

[10] M. Xie, W. Yi, T. Kirubarajan, and L. Kong, "Joint node selection and power allocation strategy for multitarget tracking in decentralized radar networks," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 729–743, Feb. 2018.

[11] X. Wang, T. Li, S. Sun, and J. Corchado, "A survey of recent advances in particle filters and remaining challenges for multitarget tracking," *Sensors*, vol. 17, no. 12, p. 2707, Nov. 2017.

[12] L. Ma, B. Wang, S. Yan, and X. Gu, "Temperature error correction based on BP neural network in meteorological wireless sensor network," *Int. J. Sensor Netw.*, vol. 23, no. 4, p. 265, 2017.

[13] G. Xiao, Q. Cheng, and C. Zhang, "Detecting travel modes using rule-based classification system and Gaussian process classifier," *IEEE Access*, vol. 7, pp. 116741–116752, 2019.

[14] W. Li, K. Xiong, Y. Jia, and J. Du, "Distributed Kalman filter for multitarget tracking systems with coupled measurements," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Jan. 8, 2020, doi: 10.1109/TSMC.2019.2960081.

[15] J. Hanyang, D. Meiling, and S. Zhilong, "An adaptive sine/cosine filtering algorithm based on speckle phase fringe orientation," *Acta Optica Sinica*, vol. 37, no. 9, 2017, Art. no. 0910001.

[16] X. Li, C. Zhao, X. Lu, and W. Wei, "DA-PMHT for multistatic passive radar multitarget tracking in dense clutter environment," *IEEE Access*, vol. 7, pp. 49316–49326, 2019.

[17] S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, "Deep learning models for real-time human activity recognition with smartphones," *Mobile Netw. Appl.*, vol. 25, pp. 743–755, Dec. 2019.

[18] J. Banfi, J. Guzzi, F. Amigoni, E. Feo Flushing, A. Giusti, L. Gambardella, and G. A. Di Caro, "An integer linear programming model for fair multi-target tracking in cooperative multirobot systems," *Auto. Robots*, vol. 43, no. 3, pp. 665–680, Mar. 2019.

[19] K. Shankar and M. Elhoseny, "Trust based cluster head election of secure message transmission in MANET using multi secure protocol with TDES," *J. Universal Comput. Sci.*, vol. 25, no. 10, pp. 1221–1239, 2019.

[20] M. S. Wei, F. Xing, and Z. You, "A real-time detection and positioning method for small and weak targets using a 1D morphology-based approach in 2D images," *Light: Sci. Appl.*, vol. 7, no. 3, p. 18006, 2018.

[21] Z. Libo, H. Tian, G. Chunyun, and M. Elhoseny, "Real-time detection of cole diseases and insect pests in wireless sensor networks," *J. Intell. Fuzzy Syst.*, vol. 37, no. 16, pp. 3513–3524, 2019.

[22] K. Muhammad, S. Khan, M. Elhoseny, S. Hassan Ahmed, and S. Wook Baik, "Efficient fire detection for uncertain surveillance environment," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 3113–3122, May 2019.

[23] X. Liang, Z. Huang, H. Qin, and Y. Liu, "GNSS multi-frequency multi-system highly robust differential positioning based on an autonomous fault detection and exclusion method," *IEEE Access*, vol. 5, pp. 26842–26851, 2017.

[24] W. Jizhou, L. Changhua, and J. Weiwei, "New multi-target tracking algorithm based on conditional random field," *J. Electron. Meas. Instrum.*, vol. 31, no. 6, pp. 909–913, 2017.

[25] C. Yang, Z. Yang, and Z. Deng, "Robust weighted state fusion Kalman estimators for networked systems with mixed uncertainties," *Inf. Fusion*, vol. 45, pp. 246–265, Jan. 2019.

[26] Z. Chen, Y. Qu, Z. Xi, Y. Bo, and B. Liu, "Efficient particle swarm optimized particle filter based improved multiple model tracking algorithm," *Comput. Intell.*, vol. 33, no. 2, pp. 262–279, May 2017.

[27] B. Li, H. Fu, D. Wen, and W. Lo, "Etracker: A mobile gaze-tracking system with near-eye display based on a combined gaze-tracking algorithm," *Sensors*, vol. 18, no. 5, p. 1626, May 2018.

[28] H. Sheng, W. Zhao, and J. Wang, "Interacting multiple model tracking algorithm fusing input estimation and best linear unbiased estimation filter," *IET Radar, Sonar Navigat.*, vol. 11, no. 1, pp. 70–77, 2017.

[29] M. Zhaoyang and Z. Wenjie, "Single lens multi target tracking algorithm based on convolution neural network," *Ship Electron. Eng.*, vol. 37, no. 12, pp. 25–28, and 38, 2017.

[30] S. Wan, "Topology hiding routing based on learning with errors," *Concurrency Comput., Pract. Exper.*, to be published.

[31] T. K. Vu, M. K. Hoang, and H. L. Le, "An EM algorithm for GMM parameter estimation in the presence of censored and dropped data with potential application for indoor positioning," *ICT Express*, vol. 5, no. 2, pp. 120–123, Jun. 2019.

[32] N. Chen, B. Rong, X. Zhang, and M. Kadoch, "Scalable and flexible massive MIMO precoding for 5G H-CRAN," *IEEE Wireless Commun.*, vol. 24, no. 1, pp. 46–52, Feb. 2017.

[33] Z. Xuewu, J. Junzhong, and Y. Y. Yao, "Insulating functional partition by searching Gaussian mixture model (GMM) using immune clone selection (ICS) algorithm," *J. Zhejiang Univ. (Eng. Ed.)*, vol. 51, no. 12, pp. 2320–2331, 2017.

[34] X. Li, Y. Wang, and G. Liu, "Structured medical pathology data hiding information association mining algorithm based on optimized convolutional neural network," *IEEE Access*, vol. 8, pp. 1443–1452, 2020.

[35] A. M. Eassa, M. Elhoseny, H. M. El-Bakry, and A. S. Salama, "NoSQL injection attack detection in Web applications using RESTful service," *Program. Comput. Softw.*, vol. 44, no. 6, pp. 435–444, Nov. 2018.

[36] M. M. Alves, L. Pirmez, S. Rossetto, F. C. Delicato, C. M. de Farias, P. F. Pires, I. L. dos Santos, and A. Y. Zomaya, "Damage prediction for wind turbines using wireless sensor and actuator networks," *J. Netw. Comput. Appl.*, vol. 80, pp. 123–140, Feb. 2017.

[37] Z. Lv and H. Song, "Mobile Internet of things under data physical fusion technology," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4616–4624, May 2020.

[38] B. Li, C. Liu, and W.-H. Chen, "An auxiliary particle filtering algorithm with inequality constraints," *IEEE Trans. Autom. Control*, vol. 62, no. 9, pp. 4639–4646, Sep. 2017.

[39] Z. Lv, B. Hu, and H. Lv, "Infrastructure monitoring and operation for smart cities based on IoT system," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 1957–1962, Mar. 2020.

[40] Z. T. Hu, J. Zhang, and Y. M. Hu, "Multi-sensor ensemble Kalman filtering algorithm based on metropolis-hastings sampling," *Tien Tzu Hsueh Pao/acta Electron. Sinica*, vol. 45, no. 4, pp. 868–873, 2017.

[41] H. He, L. Shao, H. Li, W. Pan, B. Luo, X. Zou, and L. Yan, "SNR enhancement in phase-sensitive OTDR with adaptive 2-D bilateral filtering algorithm," *IEEE Photon. J.*, vol. 9, no. 3, pp. 1–10, Jun. 2017.

[42] G. Ying and G. Sixiu, "Zero attracting sign subband adaptive filtering algorithm," *Commun. Technol.*, vol. 52, no. 2, pp. 291–297, 2019.

[43] L. Wu, C.-H. Chen, and Q. Zhang, "A mobile positioning method based on deep learning techniques," *Electronics*, vol. 8, no. 1, p. 59, Jan. 2019.

[44] F. Huang, J. Zhang, and S. Zhang, "Maximum versoria criterion-based robust adaptive filtering algorithm," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 64, no. 10, pp. 1252–1256, Oct. 2017.

XIANZHEN XU was born in Qingdao, Shandong, China, in 1988. He received the bachelor's degree from the Ocean University of China, China. He currently works with the Instrumental Analysis Center, College of Chemical Science and Engineering, Qingdao University. His research interests include chemical engineering, computational chemistry, and data analysis.

ZHIYU YUAN was born in Xinxiang, Henan, China, in 2000. He received the high school degree from the Affiliated Middle School, Henan Normal University, China. He is currently pursuing the degree with the College of Engineering, Michigan State University. His research interests include artificial intelligence and big data analysis.

YANPING WANG was born in Hangzhou, Zhejiang, China, in 1969. He received the bachelor's degree from Hangzhou Dianzi University, China, and the M.B.A. degree from Tongji University, China. He is currently working with the Information Center, China Tobacco Zhejiang Industry Company Ltd. He is also a Senior Engineer. His research interests include industrial intelligence, network transmission, and system architecture.