Distributed Data Collaborative Fusion Method for Industry-University-Research Cooperation Innovation System Based on Machine Learning

Wen Li, Hai-li Xia, and Wen-hao Guo(✉)

Business School, Suzhou University of Science and Technology, Suzhou 215009, China
ling531@hotmail.com

Abstract. Computer technology and the Internet industry are developing rapidly, and the amount of data is exploding, and people are entering the era of big data. Massive data contains a lot of knowledge value, and machine learning can extract useful key information from massive data. There are many shortcomings in traditional fusion methods, in order to better process the data in machine learning, a distributed data collaborative fusion method based on machine learning and industry-university research cooperation innovation system is proposed. The method is analyzed by research method theory and method function. The method function mainly realizes the temporal and spatial fusion of data through time synchronization, delay and misalignment of uncertain data processing, data association and weighted fusion. The simulation experiment is carried out according to the design and implementation steps of the method, and the feasibility and use value of the method are verified by experiments, and the performance of this method is superior.

Keywords: Machine learning · Innovation system · Distributed data · Fusion method

1 Introduction

Machine learning is a multi-disciplinary subject involving many disciplines such as probability theory, statistics, approximation theory, convex analysis, and algorithm complexity theory. Specializing in how computers simulate or implement human learning behaviors to acquire new knowledge or skills and reorganize existing knowledge structures to continuously improve their performance [1]. In the process of machine learning, there is an innovation model of industry-university-research cooperation to improve the efficiency of machine learning. The so-called industry-university-research cooperation is the synergy and integration of the division of labor in research, education and production in terms of functions and resources. It is the docking and coupling of technological innovations, middle and downstream. Due to the reference of innovative forms, the data in the machine-learning industry-university cooperation innovation system presents a distributed construction model. And in order to better process and analyze the data, the data is processed by a cooperative fusion method. Data Collaborative Fusion is an automated information synthesis processing

S. Liu and L. Xia (Eds.): ADHIP 2020, LNICST 347, pp. 251–261, 2021.
https://doi.org/10.1007/978-3-030-67871-5_23

technology developed and developed in the 1980s. It makes full use of the complementarity of multiple data sources and the high-speed computing and intelligence of computers to improve the quality of the resulting information. Data Co-synthesis, also known as multi-sensor fusion, is based on the combination of multi-sensor information to obtain more accurate information than any single input data source to improve the effectiveness of the entire sensor system. This performance improvement comes at the cost of increased system complexity.

Nowadays, the traditional data collaborative fusion method has the method based on time series data correlation mining. This method analyzes the operation mode of the industry university research cooperation innovation system, evaluates the related lines among the three, and calculates the correlation coefficient between the three. According to the correlation coefficient, the distributed data collaborative fusion model of industry university research cooperative innovation system is constructed, and the function is solved by using generalized EM algorithm, and the effectiveness of the fusion result is analyzed considering the weight. However, the fusion effect of this method is poor, and the actual application effect is not ideal.

Below, we analyze the architecture and fusion of the fusion system, discuss the clock synchronization problem in the distributed fusion system, and propose solutions.

2 Distributed Data Cooperative Fusion Theory

This chapter introduces the theoretical basis of the distributed information fusion method for networked systems. According to the research content of this thesis, it includes the principle of level set method, the theoretical basis of clustering algorithm, and the theory of optimal estimation, which provides a theoretical basis for the research of subsequent methods. Since its introduction, DS evidence theory has been widely used in the fields of pattern recognition, multi-sensor information fusion, image recognition, and uncertainty decision-making because of its superiority in the description and processing of uncertain events. In DS evidence theory, the recognition framework X refers to the complete set of objects studied, and the elements in X are incompatible with each other and are discrete values. The horizontal method principle is a set of horizontal slices that express an n-dimensional function by using a higher level, i.e., a level set of n 1-dimensional functions. The zero level set is a curve composed of a set of functions having the same value on the surface f(x,y) expressed by a three-dimensional continuous function, that is, a set of zero level sets of $f(x,y) = 0$ [2]. Distributed data co-fusion is analogous to the human brain to analyze the information conveyed by synthetic neuronal synapses, and thus this is an adaptive complex process. It comprehensively analyzes the environmental information that can be detected by different sensors and gives a reasonable and valuable processing result. Data fusion is a multi-level, all-round analysis process that detects, combines, correlates, estimates, and combines multiple sources of data. This results in accurate state estimation and timely, comprehensive situation assessment and threat estimation. Distributed data collaborative fusion is different from traditional data information processing in the processing of data information. The essence of data fusion is that the data processed by data fusion is more massive, bulky and highly complex. At the same time, its analysis of data is reflected in the fusion of

different levels. Combined with the above definition of data fusion, target detection and tracking is a level-level fusion, which belongs to the state estimation process of the target; target recognition belongs to the estimation process. The distributed data cooperative fusion processing structure firstly processes the data source information locally, and transmits the processed data to the fusion center, thereby reducing the bandwidth required for communication and the processing pressure of the fusion center, data fusion efficiency and system reliability are relatively more concentrated.

3 Distributed Data Collaborative Fusion Method

The architecture of the distributed data collaborative fusion method is generally distributed hierarchical processing, distributed fusion processing, and completely centralized fusion. It distributes functions such as signal processing, interconnection, tracking, attribute combination, sensor management, and status assessment to multiple processors in different ways.

It can be seen from the figure that the sensor nodes are physically distributed, and each node has a local processor (NP), which performs local data fusion (level one); The information fused locally by each node NP is linked to the global processor (P), and then the hierarchical correlation and the level two and three processing are completed by him. The fusion method is the same as the fusion mode of other structures. For the distributed data cooperative fusion model, in order to obtain a comprehensive estimation of the target motion state, the existing time fusion and spatial fusion problems are processed and considered separately. That is, when a plurality of sensors distributed at different positions are used to observe moving targets, the observation values of the sensors at different times and in different spaces will be different, thereby forming a set of observations. If there is one sensor that observes the same target at n times, there may be: s*z observations, represented by set V as $V = \{V_i\}$ $(i = 0,1,\ldots,s)$; $V_i = \{V_j(k)\}$ $(k = 0,1,\ldots,z)$, where Vi represents the set of observations of the i-th sensor, and Vj(k) represents the observation of the i-th sensor at time k. These observations have corresponding sequence numbers in time and space. In practical applications, in order to obtain the target state, these two fusions are often used in combination. Time/space fusion: Time fusion of each sensor's observation set is performed to obtain an estimate of the target state of each sensor, and then the estimation of each sensor is spatially fused to obtain a final estimate of the target state. For example, track fusion. Space/time fusion: firstly, the observation values of each sensor are fused at the same time, and the target position estimates at different times are obtained, and then time fusion is performed to obtain the final state; Time and space fusion: time fusion and spatial fusion are carried out at the same time. This method has good effect and does not lose information, but it is the most difficult and suitable for large data fusion systems [3]. The accuracy and synchronization of time is a key indicator of information accuracy, which will directly affect the accuracy of tasks such as target detection and track tracking. Obviously, for distributed data structure models, sensors are distributed in different spatial locations, and each node's local processor has its own clock. How to judge whether multiple information is collected at the same time and keep time synchronization is a key issue. With reference to the relevant theory of computer architecture, some algorithms can be used to solve the problem.

3.1 Distributed Data Clock Synchronization

In order to achieve clock synchronization, time fusion of distributed data is performed by referring to both active and passive algorithms. The active algorithm is the Berkeley algorithm, in which the time server is active, it periodically asks each machine for time. Then based on these answers, calculate the average and tell all machines to dial their clocks up or down to a new value. In this way, the time server has a time daemon and its time is maintained by the administrator. The process is as follows: First, the time daemon tells his time to other machines at a certain moment and asks them their respective time; The machines then tell the time daemon the difference between their time and the time of the daemon process; finally the daemon calculates their average and tells each machine how to adjust its time. The passive algorithm is the Cristian algorithm, which instructs a computer to be a time server, and each of the other machines periodically sends a message to the time server to ask for the current time. When the time server receives the message, it will reply to the message containing the current time value as soon as possible. When the sender gets the answer, he sets his clock to T. It is necessary to consider the time it takes to send a response from the time server to the sender. A simpler method is to accurately record the time interval from when the request is sent to the time server to when the response is received. Assuming that the start time is T0 and T1, they are all measured by the same clock. Even if the sender's clock has a certain difference from the time server clock, the time interval measured is relatively accurate. In the absence of any other information, the best estimate of the message transmission time is (T0-T1)/2. When the response message arrives, the time in the message plus this value gives the current time and the estimated time of the processed message. This estimate can be further refined if you know the time of the time server interrupt processing and the processing of the message time. Let the processing time be 1, then the transmission time interval is T0-T-I, so the one-way transmission time is half [4]. Both synchronization algorithms have a time server to provide standard time, which requires time servers to have precise time to achieve time-coherent integration on distributed data.

3.2 Uncertain Data Processing with Delay and Misordering

In the process of implementing time synchronization, the data may be affected by uncertain factors such as data transmission delay or misalignment. In order to ensure the synergistic integration of distributed data in the industry-university-research cooperation innovation system of machine learning, it is necessary to deal with such uncertain data factors. A delay compensation strategy is proposed for the delay problem. At the sampling instant k, the transmission delay is q(k), and the measured output value p(k) is stored. The received measurement output u(k) is used for state estimation b(k|t) at time t + q(k). Since the delay reduces the performance of the model, the proposed linear time-delay compensation method based on estimation is to reduce the computational complexity and further reduce the negative impact of the delay on the model. Assuming that the current sampling time is k and the received data packet is p(k), and the estimation state b(k|t) is to be performed, the state prediction value b(t + 1|t) is used for the proposed linear compensation method. Based on the maximum

delay N and the current transmission delay q(k), the estimated value b(k|t)} is obtained by the compensation formula 1.

$$\mathrm{b}\langle k|t\rangle = (1 - \frac{q(k) - 1}{N})b\langle t + 1|t\rangle \tag{1}$$

The delay is converted to a form without time lag through a series of operations. For the out-of-sequence data, the distributed data needs to be reordered, that is, reordered. When the received time-stamped data packet data is $z(k_1)$, the stored signal $p_z(k)$ is recombined into: $p_z(k) = z(k-z(k))$. Looping all the data in a distributed packet for reordering purposes, with time-stamped packets meaning in the communication network. Data is transmitted from the equipment to the filter, and the filter can obtain information on data delay and packet loss. In practical applications, the measured object is affected by the correlated noise. It is assumed that the process noise w_k and the measurement noise v_k have correlation at the same sampling time. The statistical attribute relationship is:

$$E\left(\begin{pmatrix} w_k \\ v_k \end{pmatrix}\left(w_l^T (v_l)^T\right)\right) = \begin{pmatrix} Q_k\theta_{k,l} & S_k\theta_{k,l} \\ (S_k)^T\theta_{k,l} & R_k\theta_{k,l} \end{pmatrix} \tag{2}$$

The covariance is a symmetric matrix. Find similarities in noise to perform similar processing. By using the formula to delay transform and reorder the data containing noise, the data can be removed by the method of minimum error covariance to obtain the cooperative data that can be cooperatively combined.

3.3 Data Association

Tracking distributed data and finally forming an effective trajectory, the first problem that should be solved is the problem of data source information attribution, that is, the uncertainty of data source [5]. Clustering similarity calculation is the first step in data association. The similarity measure is the amount used to describe the degree of similarity between data objects [6]. Commonly used are distance-based similarity measures, kernel-based similarity measures, connectivity-based similarity measures, cosine similarity measures, and conceptual similarity measures. Let R denote a non-empty set, if any two of the elements $i = \{M_{i1}, M_{i2}, \ldots, M_{in}\}$ and $j = \{M_{j1}, M_{j2}, \ldots, M_{jn}\}$ are bounded by a certain rule and a real number d{i,j} corresponds, and satisfies: d{i,j} is not negative; d{i,j} = 0 if and only if i and j are the same sample; d{i,j} = d{j,i};$d\{i,j\} \leq d\{i,h\} + d\{h,j\}$. In cluster analysis, the similarity measure based on Euclidean distance is common in distance as a measure of similarity.

In some cases, in order to express the importance of certain variables, Weighted Euclidean distance-based similarity measures, weighted Manhattan distance-based similarity measures, and weighted Minkowski distance-based similarity measures are proposed [7]. Therefore, at the initial stage of the target track, it is necessary to perform a measurement and measurement correlation of the plurality of cycles for the measurement to ensure the correctness of the track start. The usual data association has three processes as shown in Fig. 1.

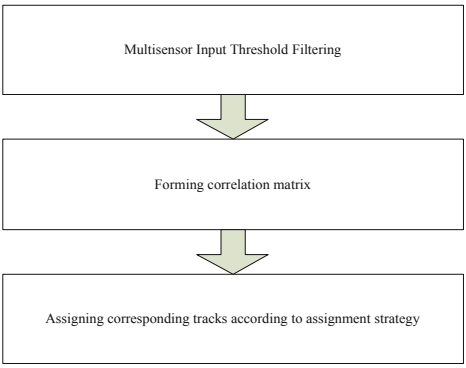

Fig. 1. Data association process

As can be seen from the figure, the data association process usually firstly filters the tracking threshold of the multi-sensor after pre-processing and data registration. Set thresholds based on prior statistical knowledge to filter out measurement data that should not appear [8]. Then, using the effective measurement of the tracking gate output to measure a track pair, an association matrix is formed, and the elements in the matrix represent the degree of association between the measurement point and the track prediction point. Finally, the measurement points with the highest degree of relevance to the predicted position are assigned corresponding tracks according to a certain assignment strategy. The assignment strategy is determined by various data association algorithms. The reasonable application of the data association algorithm is directly related to the pros and cons of the correlation effect [9].

3.4 Weighted Fusion Processing

The weighted fusion processing is performed after the correlation processing, and the distributed data weighted fusion processing flow is shown in Fig. 2.

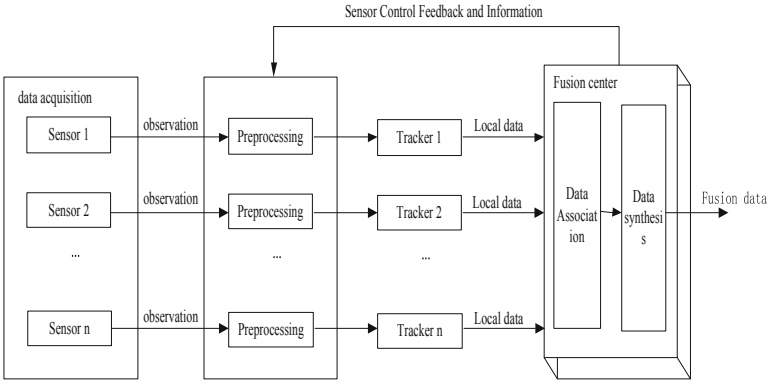

Fig. 2. Weighted fusion processing flow chart

The characteristic of the distributed fusion structure is that each branch has its own local processing system [10]. After the observation information acquired by the plurality of sensor branches is tracked in the local tracker, the target track formed by each tracker is fused at the fusion center [11]. The advantage of this configuration is that the tracking and compression processing of the measurement information is performed before the measurement information is transmitted to the fusion center, and the pressure of the communication data amount is reduced. Since local data information can be formed in each sensor branch, even if one of the roads fails, even the central system failure will not affect the formation of the track. Therefore, the reliability is high and the system survivability is strong. Increasing the tracking accuracy can be performed by time registration of the associated track and using the weighted probability to fuse the associated track [12]. Assuming that the covariance F_1 of the distributed data set 1 at the same time, and the filter value is G_1, the covariance of the data set N is F_n, and the filter value is G_n. The wave value is the covariance of the Ply radar N, and the filtered value is l JPN, then the covariance after the fusion and the fusion value are:

$$\begin{aligned} F &= (G_1^{-1}F_1 + G_2^{-1}F_2 + \ldots + G_N^{-1}F_N) \\ G &= (G_1^{-1} + G_2^{-1} + \ldots + G_N^{-1})^{-1} \end{aligned} \tag{3}$$

3.5 Implementation of Data Cooperative Fusion Method

Realizing the distributed data collaborative fusion model of the industry-university-research cooperation innovation system for machine learning. The system is tested in a real network environment, and the Hadoop distributed data processing platform of the machine learning and research cooperation innovation system model is built [13]. The distributed big data collaborative fusion method of the industry-university-research cooperation innovation system of Mapkeduce machine learning written in Java (Fig. 3):

```
FIND-MAX-CROSSING-SUBARRAY(A, low, mid, high)
1  left-sum = −∞
2  sum = 0
3  for i = mid downto low
4       sum = sum + A[i]
5       if sum > left-sum
6            left-sum = sum
7            max-left = i
8  right-sum = −∞
9  sum = 0
10 for j = mid + 1 to high
11      sum = sum + A[j]
12      if sum > right-sum
13           right-sum = sum
14           max-right = j
15 return (max-left, max-right, left-sum + right-sum)
```

Fig. 3. Algorithm pseudo code

4 Analysis of Experimental Results

The experimental demonstration uses all distributed data resources in the database of industry-university research cooperation innovation system of machine learning for data fusion processing. In order to ensure the rigor of the experiment, the traditional data fusion method, that is, the single station processing method is used as the experimental argumentation comparison, and the fusion results are statistically analyzed. The analysis results are shown in Fig. 4.

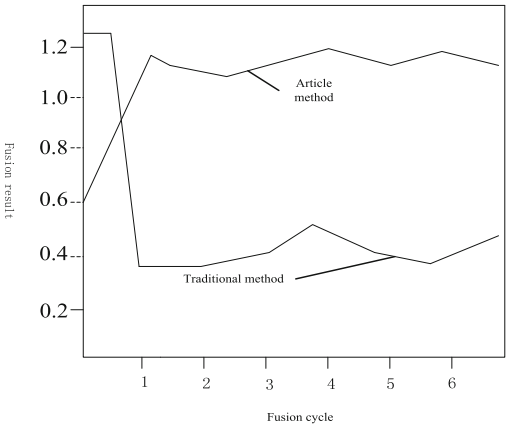

Fig. 4. Experimental results

It can be seen from the figure that in the previous cycle, the data processing mode of the single station is rapidly declining, and the data result of the data fusion mode is steadily increasing. After one cycle, the data results of the two methods are stable and the result of the fusion mode is much higher than the data result of the single station processing. Therefore, the method of transmitting the local fusion result and the processing method of the data fusion method have a large difference in the result. The system test results show that the model can guarantee the stable operation of all functions and achieve high-quality data fusion processing operations.

On the basis of the above experiments, compare the data fusion accuracy of this method and the traditional method, and the comparison results are shown in Fig. 5.

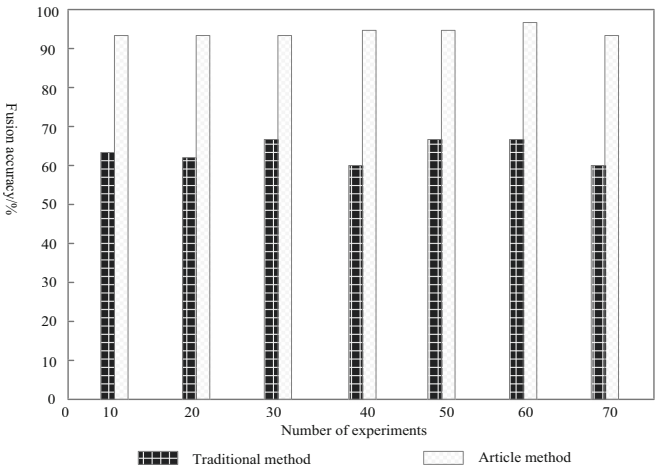

Fig. 5. Comparison of fusion accuracy

According to Fig. 5, the data fusion accuracy of traditional methods varies from 62% to 69%, while the data fusion accuracy of this method is always higher than 93%, which shows that this method can obtain accurate distributed data collaborative fusion results of industry university research cooperation innovation system. The reason is that the method mainly uses time synchronization, uncertain data processing with delay and wrong sequence, data association and weighted fusion to achieve data fusion in time and space, so it has high fusion accuracy.

On the basis of comparing the accuracy of data fusion of different methods, comparing the data fusion time of different methods, the lower the data fusion time, the higher the fusion efficiency and the better the data fusion effect. The comparison results are shown in Fig. 6.

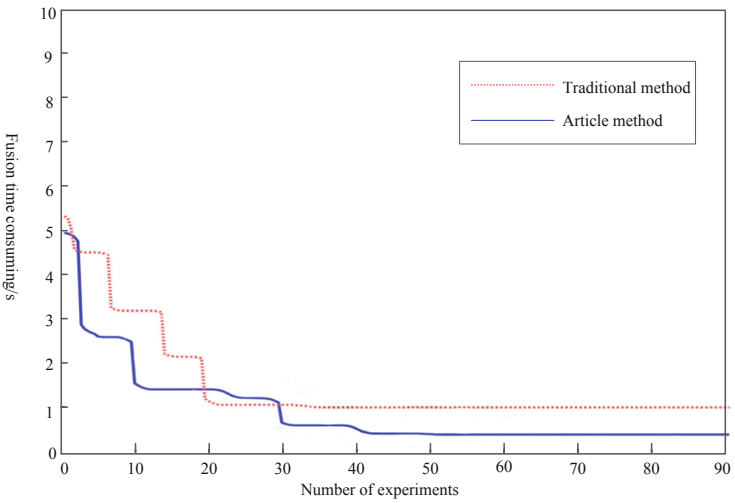

Fig. 6. Time consuming comparison of data fusion

As can be seen from Fig. 6, the data fusion time of the method in this paper varies from 0.5 s to 5.0 s, and the data fusion time of the traditional method varies from 1.0 s to 5.5 s, which indicates that the method can realize the distributed data collaborative fusion of the industry university research cooperative innovation system in a short time, and the actual application effect is good. The reason is that the method uses machine learning to process data, which improves the efficiency of data processing.

5 Conclusion

Under the background of knowledge-based economy, the traditional way of relying on factor driven and investment driven is becoming increasingly unsustainable. Relying on innovation driven is the fundamental strategy to deal with a new round of technological revolution and industrial change in the future. The intensive allocation of innovation resources and collaborative innovation among scientific research subjects will become the key support for industrial upgrading. On the premise of the difference in the goal orientation of cooperation subjects, to achieve the coupling interaction between different subjects in the cooperation network, and to maximize the limited resources into the competitive advantage, the collaborative role of industry university research cooperation subjects will become particularly prominent, It is of great significance to study the distributed data collaborative fusion method of regional industry university research cooperation innovation system. Through research, it provides a basic processing method of distributed data cooperative fusion, which provides some basic ideas for the entire data fusion strategy. The distributed data cooperative fusion method ensures the optimality of the decision result of the fusion center without increasing the amount of information transmitted and storage resources. The effectiveness of the fusion method is verified by simulation experiments and can be put into use in actual production and life.

Acknowledgments. 1. Supported by the National Natural Science Foundation of China (Grant No. 71771161);

2. Suzhou Science and Technology Program (Soft Science Project) (Grant No. SR201817).

References

1. Tang, L., Li, D., et al.: Large-scale distributed machine learning system analysis taking LDA as an example. Comput. Appl. **37**(3), 628–634(2017)
2. YuWei, S., Ma, Z., ChenYang, P., et al.: Exploration of machine learning methods with embedded expertise and experience (1): Proposal and theoretical basis of guided learning. Chin. J. Electr. Eng. **37**(19), 5560–5571 (2017)
3. LiangYi, K., JianFei, W., Liu, J., et al.: Review of parallel and distributed optimization algorithms for scalable machine learning. J. Softw. **36**(1), 109–130 (2018)
4. XuYang, T., HongBin, D., Sun, J.: Co-evolution method for feature selection. J. Intell. Syst. **12**(1), 24–31 (2017)
5. HuiDong, S., YangWen, J., GuangHeng, N., et al.: Application of machine learning in runoff prediction. Rural Water Resources Hydropower China **37**(6), 116–123 (2018)

6. Meng, Z., Yan, Z., MingDi, L., et al.: Distributed multi-sensor cooperative tracking algorithm under communication constraints. Firepower Command Control **42**(6), 6–9 (2017)
7. ZhaoMing, L., WenGe, Y., Dan, D., et al.: Distributed cooperative navigation filtering algorithm for multi-satellite cooperative targets. J. Beijing Univ. Aeronaut. Astronaut. **44**(3), 462–469 (2018)
8. Lu, M., Liu, S.: Nucleosome positioning based on generalized relative entropy. Soft. Comput. **23**(19), 9175–9188 (2018). https://doi.org/10.1007/s00500-018-3602-2
9. Fu, W., Liu, S., Srivastava, G.: Optimization of big data scheduling in social networks. Entropy **21**(9), 902 (2019)
10. Ma, H., Xie, H., Brown, D.: Eco-driving assistance system for a manual transmission bus based on machine learning. IEEE Trans. Intell. Transp. Syst. **19**(2), 572–581 (2018)
11. Xiao, F.: Multi-sensor data fusion based on the belief divergence measure of evidences and the belief entropy. Inf. Fusion **46**(2), 45–57 (2018)
12. Bader, K., Lussier, B., Sch, N.W.: A fault tolerant architecture for data fusion: a real application of Kalman filters for mobile robot localization. Robot. Auton. Syst. **88**(10), 11–23 (2018)
13. Chen, F.C., Jahanshahi, R.M.R.: NB-CNN: deep learning-based crack detection using convolutional neural network and Naïve Bayes data fusion. IEEE Trans. Industr. Electron. **65**(99), 4392–4400 (2018)