

Retraction

Retracted: A Data Acquisition and Processing Method for Edge Computing Robotic Arm Behavior Recognition

Wireless Communications and Mobile Computing

Received 1 August 2023; Accepted 1 August 2023; Published 2 August 2023

Copyright © 2023 Wireless Communications and Mobile Computing. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

- (1) Discrepancies in scope
- (2) Discrepancies in the description of the research reported
- (3) Discrepancies between the availability of data and the research described
- (4) Inappropriate citations
- (5) Incoherent, meaningless and/or irrelevant content included in the article
- (6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

- [1] Y. Zhang, Q. Xiao, X. Liu, Y. Wei, C. Chu, and J. Xue, "A Data Acquisition and Processing Method for Edge Computing Robotic Arm Behavior Recognition," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 1170373, 12 pages, 2022.

Research Article

A Data Acquisition and Processing Method for Edge Computing Robotic Arm Behavior Recognition

Yinhuan Zhang ^{1,2}, Qinkun Xiao ^{1,3}, Xing Liu ³, Yongquan Wei,⁴ Chaoqin Chu,¹ and Jingyun Xue^{1,2}

¹School of Mechatronic Engineering, Xi'an Technological University, Xi'an, 710021 Shaanxi, China

²Weinan Vocational & Technical College, Civil & Architectural Engineering, Weinan, 714000 Shaanxi, China

³Electronic Information Engineering, Xi'an Technological University, Xi'an, 710021 Shaanxi, China

⁴CRRC Tangshan Co., Ltd, Tangshan, 063000 Hebei, China

Correspondence should be addressed to Qinkun Xiao; 161847262@masu.edu.cn and Xing Liu; liuxing@stxatu.edu.cn

Received 19 April 2022; Revised 21 June 2022; Accepted 10 July 2022; Published 9 August 2022

Academic Editor: Chia-Huei Wu

Copyright © 2022 Yinhuan Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Edge computing refers to the use of an open platform on the side close to the object or data source to integrate network, computing, storage, and core application functions to provide the latest nearby services. With the development of edge computing, the cost of data acquisition has been reduced, and the efficiency has been improved. However, at present, there is no in-depth research on edge computing for robot arm behavior recognition. This paper aims to study the data acquisition and processing methods of robotic arm behavior recognition through edge computing technology. A gesture recognition method based on Cauchy distribution and grey correlation threshold is proposed, which improves the efficiency of data processing and has great research significance. In edge computing, the use of Cauchy distribution processing is more impressive; compared with empirical distribution, the algorithm optimization can reach at least 10%. Experiments show that the static gesture recognition method used in this paper is simple and high in recognition and has good robustness and the accuracy rate can basically reach more than 90%. In the case of different threshold values, when the gray correlation threshold is 0.75, the MAE value reaches the minimum value, which means that the gap between the predicted score and the actual score is the smallest, which means that the predicted result is accurate, which can prove that the recommendation of the algorithm has relatively superior performance.

1. Introduction

The society is gradually entering the era of “big data”, and with the advent of cloud computing, its ability to operate and use big data collection is also increasing. Cloud computing is actively supported by advantages including low operating costs, low weight, and ease of use and maintenance. The cloud computing industry is actively developing in China. Edge computing was initially the primary technology to address 5G latency, but has recently been introduced into new areas such as IoT and the Internet. The problem of cloud computing can also be solved with the help of modern information technology. The intelligent services provided by advanced computer technology meet key requirements such

as flexible connection, real-time operation, data upgrade, software intelligence and security, and data basic processing. Robotic arms are the most widely used robots. It exists in the construction, industrial, healthcare, entertainment, military, electronics, manufacturing, and space exploration industries. Robotic arms come in many different forms, but they all share common characteristics. The ability to select a course of action and precisely point-and-shoot in three-dimensional (or two-dimensional) space. The traditional mechanical arm can only perform simple repetitive actions such as translation and grabbing, but when the mechanical arm is combined with the information system, it can complete complex operations such as cooking and wine mixing. At this time, it is necessary to identify and analyze the

complex motion behavior of the manipulator. However, the traditional data acquisition and processing methods cannot analyze the data quickly, which leads to low efficiency. Therefore, it is necessary to collect and process the data of manipulator behavior recognition based on edge calculation.

Aiming at the data acquisition and processing of the manipulator, this paper has made great innovations in data acquisition and uploading. We first select the optimal threshold value, fully considering the accuracy of the recommendation achieved by the threshold value and the time-consuming situation and also consider the role of the threshold value in the subsequent selection of the number of neighbors. The optimal threshold is selected to realize the clustering of gray projects, which makes the internal common features of the projects prominent, and the external differences of the projects are relatively large. Calculating the similarity of items on the obtained clusters, and according to the fixed number of neighbors, sort the calculated similarities from large to small in turn and select a fixed number of neighbors. The scoring prediction and recommendation of the item are completed, and the performance of the recommendation algorithm is verified experimentally. We have done in-depth research on static gesture recognition based on multiparameter features from image acquisition to gesture recognition. The RGB-D image information is obtained by using the Kinect depth sensor, and then the gesture image is segmented by the adaptive depth threshold fused with skin color, and the corresponding preprocessing is performed to extract the outline of the gesture.

2. Related Work

With the emergence and popularity of Internet of Things (IoT) cloud services, edge computing, a new technology that requires data processing over the Internet, has emerged. Chakraborty proposed a local pattern descriptor in higher-order derivative space for face recognition. His proposed descriptor significantly reduced the extraction and matching time, while the recognition rate of the descriptor is almost comparable to the existing state-of-the-art methods [1]. Taleb's paper presented a survey of mobile edge computing (MEC), outlined current standardization activities, and further elaborated on the challenges of open research [2]. Jiang proposed a robust trajectory tracking control method for robotic arms based on H control theory, and the field operation test further verified the engineering practicability of the control method in macro and micro aspects [3]. For the rapid picking of lilies, Jiang designed a mechanical arm picking structure scheme, using a system consisting of an end effector, a manipulator, and a control system. The kinematics and picking experiments of the robotic arm were carried out in the experimental field in the natural environment through the robotic physics machine platform. The results show that the position error of the robot arm from the end of the arm is less than 12 mm, and the picking success rate is 83.33% [4]. The related research did not pay attention to saving the data of the manipulator, and even did not simulate the trajectory of the manipulator.

Maffezzoni's paper provided a realistic phase-domain modeling and simulation approach for oscillator arrays that is able to account for associated device non-idealities. This model was used to study the associative memory performance of an array consisting of resonant LC oscillators [5]. To practically implement brain-like computing in scalable physical systems, Kumar investigated a network of coupled micromachined oscillators. He performed numerical simulations of this all-pair fully coupled nonlinear oscillator array in the presence of randomness and demonstrated its ability to synchronize and store information with relative phase differences when synchronized [6]. Liu proposed an open-source face recognition method with deep representation called VIPLFaceNet, which is a 10-layer convolutional neural network consisting of 7 convolutional layers and 3 fully connected layers. Compared to the famous AlexNet, VIPLFaceNet only needed 20% training time and 60% testing time, but combined with actual LFW facial recognition metrics, it achieved a 40% error reduction. Their VIPLFaceNet achieved an average accuracy of 98.60% on LFW using a single network [7]. Biyani presented a new software package called Focus, which provided the functionality needed to remotely monitor the progress of data collection and data processing, and the rapid detection of any errors that may occur greatly increased the productivity of electron microscopy recording sessions [8]. However, none of them solve practical problems such as the redundancy of edge computing in data processing, and the following content will conduct in-depth research on these problems.

3. Edge Computing System Design and Data Sampling Method

3.1. Edge Computing System Design. The single-user scenario is shown in Figure 1.

FIGURE 1: Single-user scenario.

Whether it is cloud, fog, or edge computing, it is only a method or mode to realize the computing technology needed by the Internet of Things, intelligent manufacturing, etc. On the basis of a single system, a reserved sensor interface is used to complete the realization of the sensor acquisition function. As shown in Figure 2, by connecting the sensors that meet the requirements of the interface hardware with the edge device, the software is used to complete the preprocessing of the collected data, thereby forming the underlying data acquisition module. This can not only expand the scope of application of the sensor, but also ensure that the collected data is uploaded to the cloud in time, shortening the delay of data processing and feedback [9].

The architecture of edge computing includes four domains: equipment, network, data, and application. Platform providers mainly provide hardware and software infrastructure in network interconnection (including bus), computing power, data storage, and application. In addition to the collection of actual physical data, with the continuous development of informatization today, a large amount of data existing on the Internet is also worthy of attention and utilization. Therefore, for the data that can be accessed on the network, especially those web page data presented

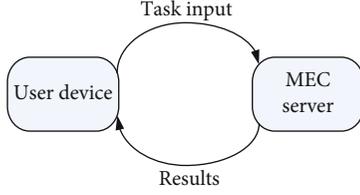


FIGURE 1: Single-user scenario.

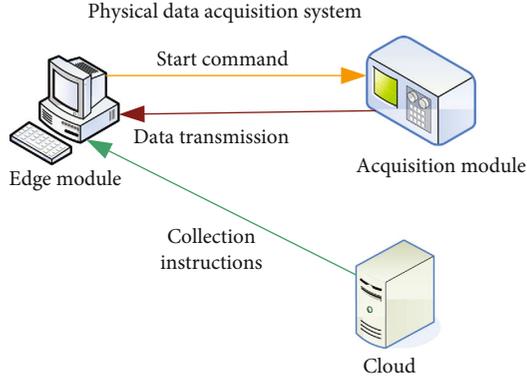


FIGURE 2: Physical data acquisition system.

in HTML and other formats, people will use various methods to download the web page codes and the data behind them and process them. As shown in Figure 3, the system utilizes the characteristics of a large number of edge devices and low energy consumption and can establish a multi-threaded or long-term monitoring data capture system by providing it with a crawler program. For example, users can carry sensors through the edge device network to monitor the air quality of a certain area and at the same time use web crawler technology to query the indexes of various local social activities, such as factory operating conditions and regional weather. In this way, it is compared with the collected air parameters to analyze the factors that affect the local air quality.

The rapid development of global smart phones has promoted the development of mobile terminals and “edge computing.” The intelligent society with the internet of things and the perception of everything is accompanied by the development of the Internet of Things, and the edge computing system has emerged accordingly. In the edge computing system, the platform needs to be built by the cloud computing center and edge devices. The advantage of this architecture is that when the bottom layer needs to call the data analysis function, the edge device can complete the timely feedback, shorten the response time, and ensure the real-time response characteristics of edge computing [10]. For the Internet of Things, breakthroughs in edge computing technology mean that many controls will be implemented through local devices without being handed over to the cloud, and the processing will be completed at the local edge computing layer. This will undoubtedly greatly improve processing efficiency and reduce the load on the

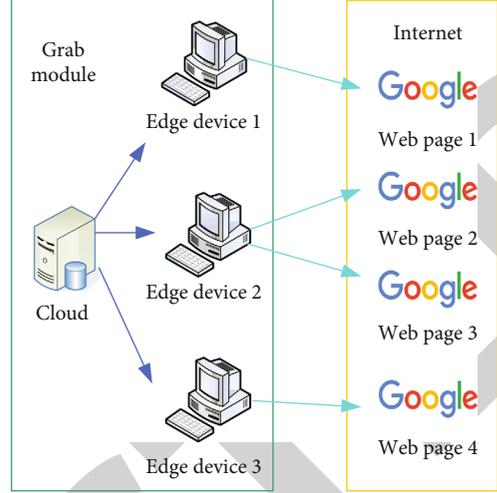


FIGURE 3: Network data capture system.

cloud. Due to being closer to the user, it can also provide users with a faster response and solve their needs at the edge.

3.2. Pattern Recognition Methods

3.2.1. Host Component Analysis Method. Principal component analysis is a statistical method. Through orthogonal transformation, a group of variables that may be related are converted into a group of linearly unrelated variables, and the converted variables are called principal components. The traditional principal component analysis algorithm (PCA) is a method to minimize measurement features and is an important tool for measurement analysis. It has a predictable learning path [11]. Now PCA is continued to be derived for the mean least squares error of the object.

$$\lambda_y = \frac{1}{n} \sum_{j=1}^n y_j. \quad (1)$$

$$y_j' = y_j - \lambda_y \quad (2)$$

The sample matrix after zero mean processing is written as Y' , and $Y' = \{y_1', y_2', \dots, y_n'\}$. After the sample zero-mean centering, its autocorrelation function can be expressed as

$$B_y = Y' Y'^T. \quad (3)$$

Let the optimal transformation vector be $Cd \in R^l$, and find the eigenvalues and eigenvectors of B_y :

$$B_y d = \eta d. \quad (4)$$

The largest eigenvector obtained from this characteristic formula is the optimal PCA transform vector [12]. In the PCA dimensionality reduction process, it is usually necessary to manually adjust the reduced size [13]. The transformation matrix after dimensionality reduction can be

written as

$$D = \{d_1, d_2, \dots, d_j, \dots, d_k\}, d_j \in R^l. \quad (5)$$

3.2.2. Kernel Principal Component Analysis. If the purpose of PCA is to attenuate nonlinear correlations between a given data set ($Y = \{y_1, y_1, \dots, y_j, \dots, y_n\}, y_i \in R^l$), their covariances can be represented on a linear feature space F instead of the nonlinear representation in the original input space [14].

$$B^F = \frac{1}{n} \sum_{i=1}^n \Phi(y_i) \Phi(y_i)^T. \quad (6)$$

It is assumed that the mapping of the feature vector in the high-dimensional space is also zero-mean, that is, $\sum_{k=1}^n \Phi(y_k) = 0$ and $\Phi(\bullet)$ are nonlinear mapping functions that map the input feature vector to the high-dimensional space F. Then, diagonalize the covariance matrix:

$$\mu v = B^F v. \quad (7)$$

Next, the eigenvalues and eigenvectors are solved by solving the characteristic formula. The specific content of $B^F v$ can be deduced according to Formula (6):

$$B^F v = \left(\frac{1}{n} \sum_{i=1}^n \Phi(y_i) \Phi(y_i)^T \right) v = \frac{1}{n} \sum_{i=1}^n \langle \Phi(y_i), v \rangle \Phi(y_i). \quad (8)$$

In the formula, $\langle u, v \rangle$ represents the inner product between u and v . The formula shows that all eigenvectors v of $\mu \neq 0$ are composed of $\Phi(y_1), \dots, \Phi(y_n)$, so we can use $v = \sum_{j=1}^n$ to represent the eigenvector, and Formula (7) can also be written as

$$\mu \langle \Phi(y_k), v \rangle = \langle \Phi(y_k), B^F v \rangle, k = 1, 2, \dots, n. \quad (9)$$

Combining (8) and (9), we can get

$$\mu \sum_{i=1}^n \beta_i \langle \Phi(y_k), \Phi(y_i) \rangle = \frac{1}{n} \sum_{i=1}^n \beta_i \left\langle \Phi(y_k), \sum_{j=1}^n \Phi(y_j) \right\rangle \langle \Phi(y_j), \Phi(y_i) \rangle. \quad (10)$$

Now, a matrix K of size $n \times n$ is assumed, where $K_{ij} = \langle \Phi(y_i), \Phi(y_j) \rangle$; then, the left-hand side of Formula (10) can be written as

$$\mu \sum_{i=1}^n \beta_i \langle \Phi(y_k), \Phi(y_i) \rangle = \mu \sum_{i=1}^n \beta_i K_{ki} = \mu K \beta. \quad (11)$$

The right-hand side of the formula can be written as

$$\frac{1}{n} \sum_{i=1}^n \beta_i \left\langle \Phi(y_k), \sum_{j=1}^n \Phi(y_j) \right\rangle \langle \Phi(y_j), \Phi(y_i) \rangle = \frac{1}{n} \sum_{i=1}^n \beta_i \sum_{j=1}^n K_{kj} K_{ji} = \frac{1}{n} K^2 \beta. \quad (12)$$

Combining Formulas (11) and (12), we get

$$\mu n \beta = K \beta. \quad (13)$$

3.3. Cauchy Distribution Function Data Sampling and Processing. The distribution is estimated from the sample using the Cauchy distribution function, mainly the estimation of the parameters of the Cauchy distribution, and then the inverse cumulative distribution function is used for sampling [15]. A description of the Cauchy distribution function and parameter estimation and sampling methods is first given [16].

The Cauchy distribution, also known as the Cauchy-Lorentz distribution, is a continuous distribution function. If the probability density function of random variable ξ is

$$f(y, y_0, \gamma) = \frac{1}{\pi \gamma [1 + (y - y_0/\gamma)^2]} = \frac{1}{\pi} \left[\frac{\gamma}{(y - y_0)^2 + \gamma^2} \right]. \quad (14)$$

Among them, y, y_0 are constants, and $\gamma > 0$, then ξ is said to obey the Cauchy distribution with parameters γ, y_0 and denoted as $\xi \sim C(\gamma, y_0)$. The cumulative distribution function of $\xi \sim C(\gamma, y_0)$ is

$$F(y, y_0, \gamma) = \frac{1}{\pi} \arctan \left(\frac{y - y_0}{\gamma} \right) + \frac{1}{2}. \quad (15)$$

In the formula, y_0 is the position parameter that defines the location of the peak of the distribution, and γ is the scale parameter that is half width at half the maximum value. And $F_\xi(y_0) = 1/2$, and $F_\xi(y + y_0) = 3/4$, the former formula states that y_0 is the median of $C(\gamma, y_0)$, while the latter formula states that $\gamma + y_0$ is the 3/4 quantile of $C(\gamma, y_0)$.

In particular, the Cauchy distribution with parameter $y_0 = 0, \gamma = 1$ is called the standard Cauchy distribution, denoted by $\xi \sim C(\gamma, y_0)$. Its probability density function is

$$f(y; 0, 1) = \frac{1}{\pi(1 + y^2)}. \quad (16)$$

Its cumulative distribution function is

$$\exp(y_0 i t - \gamma |t|). \quad (17)$$

The properties of the Cauchy distribution: the expectation and variance are undefined, the median or mode is y_0 , the entropy value is $\ln 4\pi x$, and the eigenfunction is $\exp(y_0 i t - \gamma |t|)$.

It can be seen from the characteristics of the Cauchy distribution that the Cauchy distribution is a continuous distribution function with neither expectation nor variance. Therefore, the classical parameter estimation method cannot be used to estimate the parameters of the Cauchy distribution [17].

For continuous distribution functions, the inverse cumulative distribution function can be used for sampling.

Supposing the distribution function of the sample random variable X is $F(X)$, and the function $F(X)$ is continuous. Its inverse cumulative distribution function is also its inverse function, denoted as $F^{-1}(z)$, $0 \leq z \leq 1$.

In Theorem 1, if the random variable R is uniformly distributed on $(0,1)$, that is, $R \sim U(0,1)$, then $Y = F^{-1}(R)$ is the inverse function of $F(Y)$.

From the cumulative distribution function formula of the Cauchy distribution, its inverse cumulative distribution function can be obtained as

$$F_{-1}(q; y_0, \eta) = y_0 + \eta \tan \left(\pi \left(q - \frac{1}{2} \right) \right). \quad (18)$$

It can be known from Theorem 1 that if people want to generate a random variable number that obeys $F(Y)$, first generate a random variable number u that obeys $U(0,1)$ uniform distribution, and finally calculate $F^{-1}(u)$.

4. Edge Algorithm Simulation Test Experiment and Data Processing Analysis

4.1. *Edge Algorithm Simulation Test Experiment.* To test the performance of the algorithm, experiments were performed using the following four test functions:

Rosenbrock:

$$g(y) = \sum_{i=1}^m \left[100(y_{i+1} - y_i^2)^2 + (y_i - 1)^2 \right]. \quad (19)$$

In the formula, $-10 \leq y_i \leq 10$, $\min(g) = g(1, \dots, 1) = 0$.

SumCan:

$$g(y) = - \left\{ 10^5 + \sum_{i=1}^n |x_i| \right\}^{-1}. \quad (20)$$

In the formula, $x_i = y_i$, $x_i = x_{i-1} + y_i$, $i = 2, \dots, d$, $-0.16 \leq y_i \leq 0.16$, $\min(g) = g(0, \dots, 0) = -10^5$.

Sphere:

$$g(y) = \sum_{i=1}^n y_i^2. \quad (21)$$

In the formula, $-100 \leq y_i \leq 100$, $\min(g) = g(0, \dots, 0) = 0$.

Schwefel2.22:

$$g(y) = \sum_{i=1}^n |y_i| + \prod_{i=1}^n |y_i|. \quad (22)$$

In the formula, $-10 \leq y_i \leq 10$, $\min(g) = g(0, \dots, 0) = 0$.

The test function Rosenbrock is a classical function, and it is a complex optimization problem. Its shape is parabolic and narrow, but the position of the global optimal solution is in the smoothest flat valley of this parabolic shape, and most of the algorithms can search this parabolic canyon, but it is difficult to find the global optimal solution [18].

TABLE 1: Optimal solution and optimal value of test function.

Test function	Optimal solution	The optimal value
Rosenbrock	$y = (1, \dots, 1)$	0
SumCan	$y = (1, \dots, 1)$	-10^5
Sphere	$y = (0, \dots, 0)$	0
Schwefel2.22	$y = (0, \dots, 0)$	0

The test function SumCan is shaped like an inverted spire, and the optimal position is at this inverted spire, and other positions are relatively flat. Therefore, the position of the global optimal solution searched by many algorithms during optimization is very close to the optimal position of the test function, but it is very different from the optimal fitness value, which is -10^5 .

The test function Sphere is a unimodal function with simple relationships between variables. The function test Schwefel is a multimodal function, and its local optimum position is far away from its global optimum position because its local optimum value is also very small. Thus, when it converges to the local optimum, it is easy to mistakenly think that it has converged to the global optimum. At the same time, it is difficult for the algorithm to get rid of the local optimum that has been trapped, and it is difficult to achieve the level of global optimization. Therefore, this test function is often used to analyze the execution performance of algorithms out of local optima. The test function Rastrigin is a multimodal function. Similar to the test function Schwefel, it is used to test the execution ability of the algorithm to get rid of the local optimum. The difference is that its local optimum is located near the global optimum. So it is easy to fall into local optimum. The test function Griewank is a multimodal function with many local optimal points, and each dimension variable has a strong interaction relationship. Due to the strong interaction and influence of each dimension variable, the number of local optimal points will increase with the increase of dimensions and increase sharply, thus making it extremely easy for the algorithm to fall into local optimum points and not easily get out of it. Therefore, the test function is often used to test the ability of the algorithm to get rid of local optima and the performance of the optimization problem for high-dimensional variables. The optimal solution and optimal value of the test function are shown in Table 1.

The simulation results of the test function in the marginal distribution are shown in Figure 4.

The population size is 2000, the dimension is 10, and the selection rate is 0.5. Some individuals are selected as part of the new population by truncation selection and roulette selection. The mutation operator is used in the algorithm, and the mutation rate is 0.05. The algorithm stops when any of the following conditions are met. The conditions are as follows: ① The difference between the optimal values of the evolution of the adjacent two generations of the algorithm within 25 consecutive generations is less than $1e-6$; ② the optimal value is found; and ③ the maximum fitness evaluation times are 300,000. Each function was run 50 times independently in different test environments. In

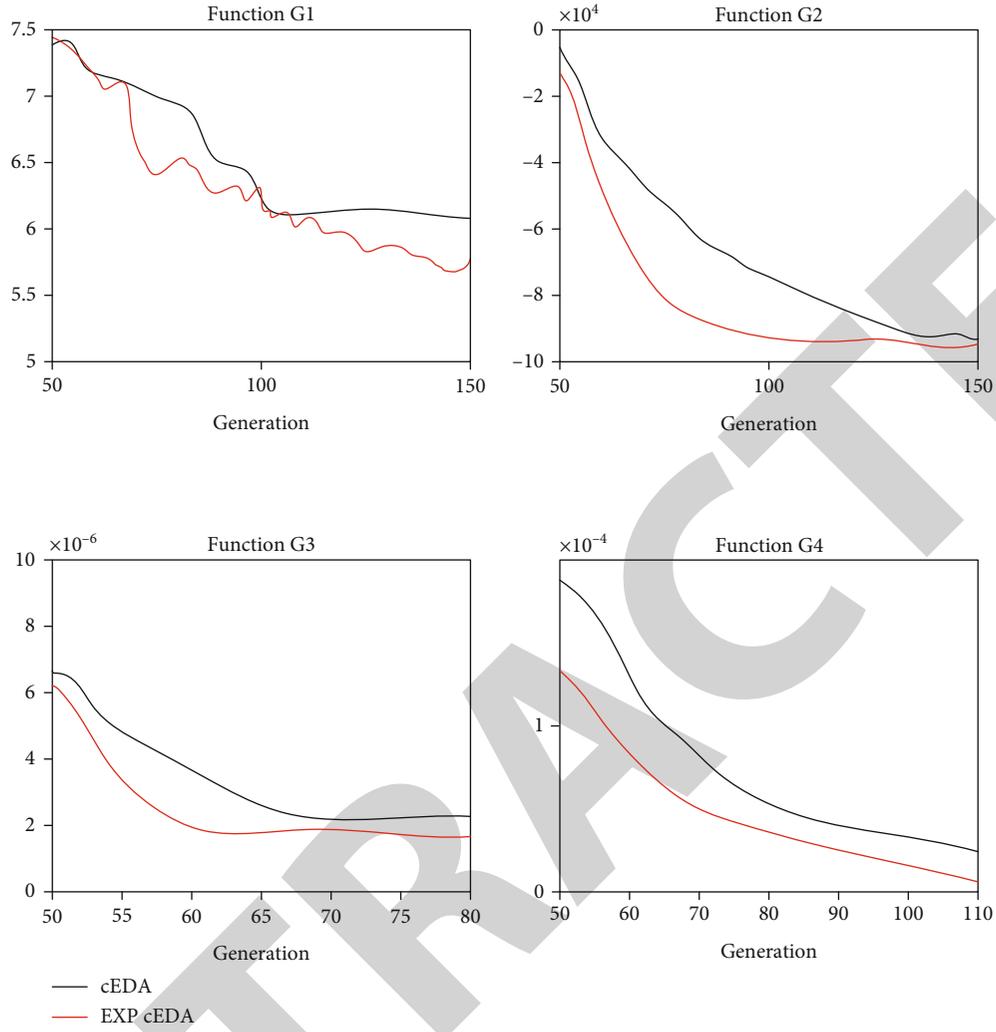


FIGURE 4: Simulation results of the test function in the marginal distribution.

TABLE 2: Comparison of copulaEDA simulation results of two marginal distribution functions.

Test function	Marginal distribution	Average value	Variance	Minimum value	Maximum value
Rosenbrock	Empirical distribution	6.5534	0.3746	5.9452	8.1534
	Cauchy distribution	5.4324	0.2634	4.4562	5.1256
SumCan	Empirical distribution	-8.3573e+04	9.4638e+03	-9.7832e+04	-5.5346e+04
	Cauchy distribution	-8.9805e+04	1.0372e+04	-9.3278e+04	-5.5527e+04
Sphere	Empirical distribution	4.6734e-08	2.7839e-08	7.9452e-09	1.4627e-07
	Cauchy distribution	2.4537e-08	1.9875e-08	1.5848e-09	9.7462e-08
Schwefel2.22	Empirical distribution	4.6738e-06	1.7839e-06	1.4268e-06	8.4528e-06
	Cauchy distribution	7.3279e-07	3.0945e-07	2.4837e-07	1.5371e-06

addition, in the copula distribution estimation algorithm of the Cauchy distribution probability model, the parameter estimation adopts the quantile estimation method for simulation experiments.

From the experimental results in Table 2, it can be seen that for the four test functions, the copulaEDA whose mar-

ginal distribution is a Cauchy distribution function is better than the copulaEDA with an empirical distribution function. For the test function Rosenbrock, it can be seen from Table 2 that whether the empirical distribution function is used as the marginal distribution or the Cauchy distribution function is used as the marginal distribution function, the

TABLE 3: Gesture recognition results in different environments.

Serial number	Glare		Low light		Background distraction	
	Recognition	Accuracy rate/%	Recognition	Accuracy rate/%	Recognition	Accuracy rate/%
1	499/500	99.8	499/500	99.8	498/500	99.6
2	499/500	99.8	489/500	97.8	494/500	98.8
3	489/500	97.8	479/500	95.8	496/500	99.2
4	469/500	93.8	469/500	93.8	490/500	98.0
5	499/500	99.8	459/500	91.8	459/500	91.8
6	498/500	99.6	449/500	89.8	498/500	99.6
7	489/500	97.8	439/500	87.8	459/500	91.8
8	479/500	95.8	459/500	91.8	497/500	99.4

TABLE 4: Recognition results of different operators and gesture changes.

Serial number	Different operators		Gesture rotation		Gesture zoom	
	Recognition	Accuracy rate/%	Recognition	Accuracy rate/%	Recognition	Accuracy rate/%
1	498/500	99.6	499/500	99.8	498/500	99.6
2	499/500	99.8	489/500	97.8	494/500	98.8
3	489/500	97.8	479/500	95.8	496/500	99.2
4	469/500	93.8	469/500	93.8	490/500	98.0
5	499/500	99.8	459/500	91.8	458/500	91.6
6	498/500	99.6	459/500	91.8	498/500	99.6
7	489/500	97.8	459/500	91.8	459/500	91.8
8	479/500	95.8	459/500	91.8	497/500	99.4

simulation results show that the optimization effect of the copula distribution estimation algorithm is relatively poor. Relatively speaking, the copula distribution estimation algorithm of the Cauchy distribution probability model is relatively better than the copula distribution estimation algorithm of the empirical distribution probability model in terms of optimization performance and tending to stability [19].

For the test function SumCan, it is also a difficult problem in the distribution estimation algorithm to continue to optimize. From the results in Table 2, it can be seen that the optimization results of the two algorithms are not good and the difference is not large. For the test function Sphere, it can be seen from the experimental results that the optimization result of the copula distribution estimation algorithm of the Cauchy distribution probability model on this function is better than the copula distribution estimation algorithm of the empirical distribution probability model. For the test function Schwefel, the copula distribution estimation algorithm of the Cauchy probability model is obviously superior to the Clayton copula distribution estimation algorithm of the empirical distribution probability model in terms of mean, variance, minimum and maximum values during optimization. After careful calculation, the optimization rate of the Cauchy model has reached more than 10%.

4.2. Gesture Recognition Robustness Verification. In the current vision-based gesture recognition, most of them use

color cameras to obtain image information, while the color camera-based visual gesture recognition system has high requirements on the environment. And it is easily affected by changes in illumination, resulting in the inability to accurately segment gestures part, which ultimately affects the result of gesture recognition; in skin-color-based gesture detection, it is unavoidable that it will be affected by the skin-like environment and its own skin color and gestures cannot be extracted from complex backgrounds. This paper uses the Kinect depth camera to obtain RGB-D image information. The infrared camera group it contains can project and receive infrared light to obtain depth data within the visible environment. Changes in ambient lighting will not affect the acquisition of depth data, and depth information can be used for gesture recognition even in dark environments [20]. Through the depth threshold gesture segmentation fused with skin color information, the interference of non-skin color environmental factors in the effective gesture area can be excluded, thereby improving the accuracy of gesture segmentation. In the calculation of the gesture feature parameters, the judgment of the effective convex defect and the ratio of the gesture area to the minimum circumscribed circle area are all in the form of ratio, which makes the gesture recognition have good adaptability; the effective convex defect feature angle and the effective vertex itself have scaling and rotation invariance, correct recognition of gesture rotation, and scaling. In order to verify the robustness of the static gesture recognition method, gesture recognition experiments are carried out under strong light, weak light, and

complex backgrounds. The experimental results are shown in Table 3.

The static gesture recognition method adopted in this paper can well overcome the influence of different light intensities and can also correctly recognize gestures in the presence of background interference [21]. The experimental results of gesture recognition under different operators and gesture rotation and zoom conditions are shown in Table 4. Although different operators have differences in expressing gestures, or excessive zooming in the process of gesture changes weakens gesture features, it will eventually lead to identify the wrong situation, but can get a good recognition effect within the effective recognition range.

An in-depth study on static gesture recognition based on multiparameter features from image acquisition to gesture recognition have been done [22]. The RGB-D image information is obtained by using the Kinect depth sensor, and then the gesture image is segmented by the adaptive depth threshold fused with skin color, and the corresponding preprocessing is performed to extract the outline of the gesture. Experiments show that the static gesture recognition method used in this paper is simple and high in recognition and has good robustness, and the accuracy rate can basically reach more than 90%.

4.3. Data Processing Experimental Analysis

4.3.1. *Experimental Process.* The experimental flow chart in Figure 5 clearly describes the entire experimental process.

First, selecting the optimal threshold value, fully consider the accuracy of the recommendation achieved by the threshold value and the time-consuming situation, and considering the effect of the threshold value in the subsequent selection of the number of neighbors. Selecting the optimal threshold to achieve gray project clustering makes the internal common features of the project prominent, and the external differences between the projects are large [23]. Calculating the similarity of items on the obtained clusters, and according to the fixed number of neighbors, sort the calculated similarities from large to small in turn, and select a fixed number of neighbors. The scoring prediction and recommendation of the item are completed, and the performance of the recommendation algorithm is verified experimentally.

4.3.2. *Performance Evaluation Criteria.* The performance of the algorithm is verified by experimental simulation. During the experiment, other recommendation algorithms are used to compare and analyze the algorithm. The movie data set is used, and the recommendation algorithm is used to operate the data set. And the obtained simulation results can test the efficiency and accuracy of the algorithm we study.

The evaluation method of the pros and cons of the recommendation algorithm has also been under exploration. Nowadays, the commonly used evaluation schemes are as follows: decision support accuracy and statistical accuracy. The most commonly used is the mean absolute difference measurement method among the statistical accuracy measurement methods. Therefore, this paper uses the MAE algorithm to evaluate the recommendation system. The MAE

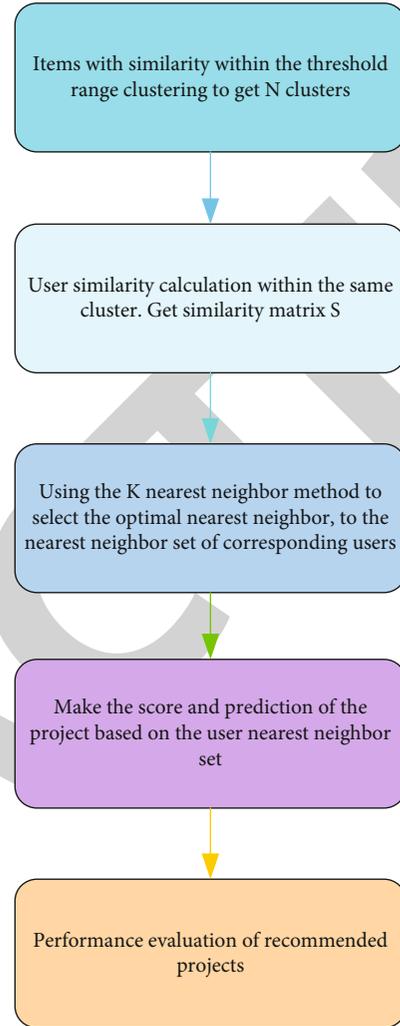


FIGURE 5: Experimental flow chart.

value represents the value of the degree of deviation between the predicted user score calculated by the algorithm and the actual score, so when the obtained value is smaller, it means that the difference between the predicted value and the actual value is smaller, so it also proves that the accuracy of the recommendation algorithm is higher, which also clarifies the effectiveness of the recommendation algorithm [24].

The calculation formula of the MAE evaluation scheme is as follows:

$$MAE = \frac{\sum_{j=1}^M |G_j - A_j|}{T}. \quad (23)$$

In Formula (23), G_j represents the predicted value of the user's rating, A_j represents the actual value of the known user's rating, and T represents the total number of rating items.

4.3.3. *Selection of Optimal Threshold for Grey Clustering.* Given a known movie data set, the classification situation has been clarified, and the original data set is subjected to

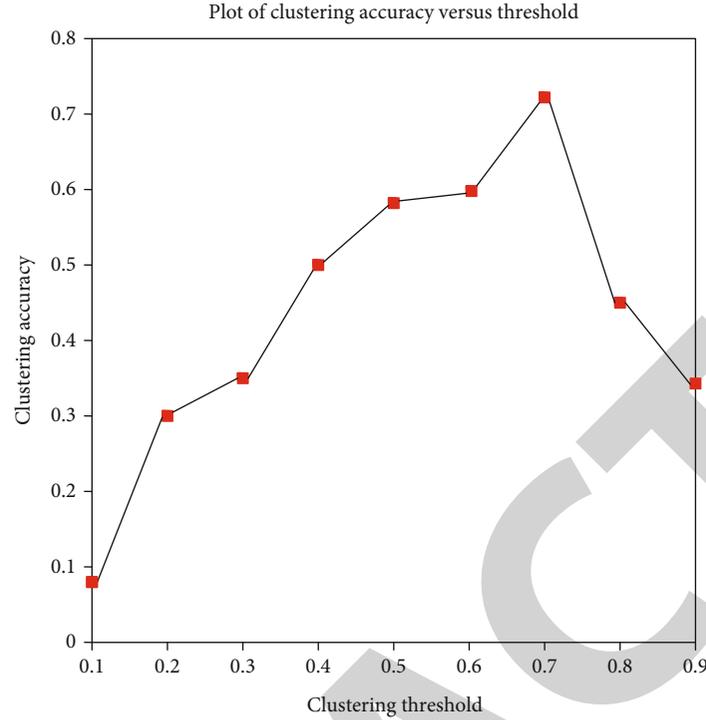


FIGURE 6: Clustering accuracy analysis.

TABLE 5: Item clustering result analysis.

Threshold value	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Number of clustering error samples	3668	3077	2561	2341	1845	1598	1178	1234	2323
Clustering accuracy	7.3	24	30.9	40.8	53.5	59.8	73.3	68.9	40.9
Number of categories	20	46	47	64	75	110	121	145	321

the gray theory to realize the project clustering process, and a new data set classification situation will be obtained, and the gray project clustering algorithm is based on thresholds are used to classify categories. Therefore, in the experimental process, the known classification situation is compared with the clustering results obtained by the gray clustering algorithm when the threshold values are different so that it can be clear that the highest accuracy obtained under a certain clustering threshold can be obtained the classification result. During the experiment, the threshold value is from 0.1 to 0.9, and the obtained clustering result is shown in Figure 6, which vividly describes the change trend of the clustering accuracy curve under the condition of different threshold selection [25].

It can be clearly seen from Figure 6 that in the process of obtaining the threshold from 0.1 to 0.7, the accuracy of the item clustering division of the gray correlation algorithm is a monotonically increasing function, and the accuracy is known to improve; when the threshold value is obtained from 0.7 to 0.9, it can be clearly seen from the vertical coordinate of the accuracy in Figure 6 that the accuracy has a downward trend. It can be seen that when the grey relational

clustering threshold is 0.7, the highest clustering accuracy can be obtained, that is, the divided item clusters are the least different from the actual known item clusters [26]. Therefore, the optimal threshold of the grey relational clustering algorithm obtained is 0.7.

Table 5 describes the number of samples of clustering errors generated under each threshold and the corresponding clustering accuracy when different clustering thresholds are selected. And the number of categories generated by clustering has made a specific numerical elaboration and comparison.

It can be seen from Table 5 that when the optimal threshold for grey relational clustering is 0.7, the accuracy of the clustering algorithm reaches 73.3%, and a total of 121 categories of movie clusters are generated. In order to further optimize the selection of the threshold, the experimental range of the threshold is selected between 0.6 and 0.8, and the obtained clustering accuracy is shown in Figure 7.

From Figure 7, we can see that the accuracy of clustering shows a clear upward trend in the process of changing the threshold from 0.60 to 0.75, while the accuracy decreases

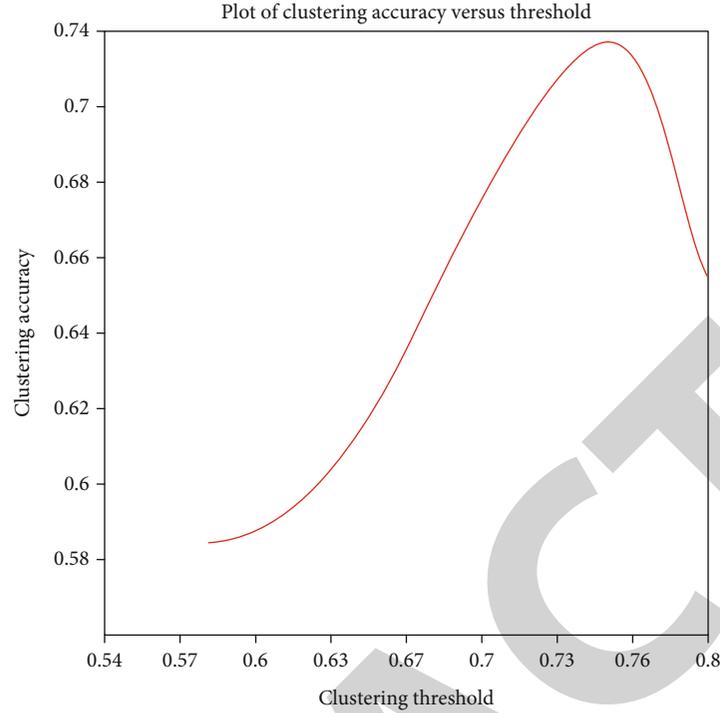


FIGURE 7: Clustering accuracy refinement analysis.

TABLE 6: Analysis of the clustering results of items with a threshold in the range of 0.61-0.69.

Threshold value	0.61	0.62	0.63	0.64	0.65	0.66	0.67	0.68	0.69
Number of clustering error samples	1567	1534	1500	1463	1422	1376	1327	1281	1234
Clustering accuracy (%)	60.2	61.1	62.1	63.1	64.2	65.0	66.5	67.5	68.7
Error (%)	1.2	2.1	1.4	2.3	3.1	2.2	2.2	2.2	2.3

TABLE 7: Analysis of item clustering results with thresholds in the range of 0.71-0.79.

Threshold value	0.71	0.72	0.73	0.74	0.75	0.76	0.77	0.78	0.79
Number of clustering error samples	1136	1098	1065	1063	1038	1028	1033	1055	1111
Clustering accuracy (%)	70.2	71.1	72.1	73.1	74.2	73.0	71.5	71.0	70.5
Error (%)	2.3	3.3	3.3	3.4	3.1	3.0	2.5	2.6	2.3

significantly when the threshold changes from 0.75 to 0.8. Tables 6 and 7 describe the clustering accuracy obtained with further refinement of the clustering threshold selection. Table 6 describes the clustering accuracy analysis when the threshold range is between 0.61 and 0.69, and Table 7 describes the accuracy when the clustering threshold is selected between 0.71 and 0.79. The number of samples corresponding to clustering errors under each threshold and the corresponding clustering accuracy can be used to determine the optimal threshold more accurately [27].

Table 7 is an analysis of the clustering results of items with a threshold in the range of 0.71-0.79.

It can be seen from Tables 6 and 7 that when the gray correlation clustering threshold is 0.75, the highest clustering accuracy can be obtained, that is, the difference between

the divided item clusters and the actual known item clusters is the least, so the optimal threshold of the obtained grey relational clustering algorithm is 0.75.

In the grey relational clustering algorithm in this paper, the clustering based on user items is completed, so in the process of comparing recommendation algorithms, it cannot be compared with the commonly used classical clustering methods that use item attributes as a reference. Therefore, in order to better study the performance of the recommendation algorithm, we introduce a gray-based collaborative filtering recommendation algorithm for comparison to verify the effectiveness of gray clustering. The ratio of the test set to the training set is 1:4 randomly divided, 20% of the test set and 80% of the training set are selected, and the number of neighbors is 40. Under different gray clustering

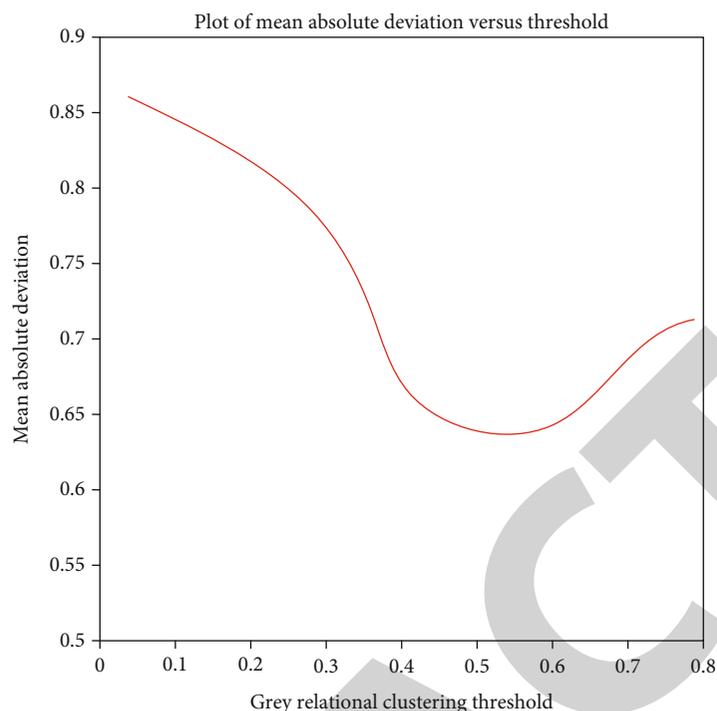


FIGURE 8: MAE values under different clustering thresholds.

thresholds, the item scores are predicted and the MAE value is obtained.

It can be seen from Figure 8 that under different threshold values, when the gray correlation threshold is 0.75, the MAE value reaches the minimum value, which means that the gap between the predicted score and the actual score is the smallest, which means that the predicted result is very accurate, so it can be proved that the recommendation algorithm has relatively superior performance. Therefore, choosing the optimal threshold value as 0.75 can achieve the best user recommendation effect.

5. Discussion

Although the method in this paper reduces the complexity of the system to a certain extent, if the system is applied in an embedded platform, the running speed is still relatively slow. How to further improve the recognition efficiency of the whole system is the focus of future research. In the process of threshold selection for grey item clustering, several experiments are carried out in this paper to obtain a fixed optimal threshold. In the future, the dynamic adaptive algorithm of item clustering threshold can be studied. In the process of finding the optimal threshold, the optimal threshold is used to achieve the highest accuracy of gray item clustering and to alleviate the influence of the number of neighbors on the selection of neighbors and the final recommendation effect.

6. Conclusions

In edge computing, the use of Cauchy distribution processing is more impressive; compared with empirical distribution, the algorithm optimization can reach at least 10%.

Experiments show that the static gesture recognition method used in this paper is simple and high in recognition and has good robustness and the accuracy rate can basically reach more than 90%. In the case of different threshold values, when the gray correlation threshold is 0.75, the MAE value reaches the minimum value, which means that the gap between the predicted score and the actual score is the smallest, which means that the predicted result is accurate, which can prove that the recommendation of the algorithm has relatively superior performance. Therefore, the optimal threshold value is selected as 0.75, which can achieve the best effect. Moreover, aiming at the behavior of the manipulator, we will have a deeper understanding of computer-related technologies and simulate a better behavior of the manipulator.

Data Availability

Data sharing is not applicable to this article as no new data was created or analyzed in this study.

Conflicts of Interest

The author states that this article has no conflict of interest.

Acknowledgments

This work is supported by the Nature Science Foundation of China (No. 6207010855) and Weinan Science and Technology Bureau in Shaanxi Province of China (No. 2020ZDYF-JCYJ-235).

References

- [1] S. Chakraborty, S. K. Singh, and P. Chakraborty, "Local directional gradient pattern: a local descriptor for face recognition," *Multimedia Tools & Applications*, vol. 76, no. 1, pp. 1201–1216, 2017.
- [2] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: a survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.
- [3] W. Jiang, G. Wu, W. Wang, J. Zhang, F. Fan, and X. Ye, "Robust motion control of live-line maintenance robot mechanical arm for high-voltage transmission line based on H theory," *Transactions of the Institute of Measurement and Control*, vol. 40, no. 3, pp. 951–967, 2018.
- [4] A. Jiang, X. Yao, M. Cheng, and J. Zhou, "Kinematics analysis and experiment of a lily picking mechanical arm," *The Journal of Engineering*, vol. 2018, no. 16, pp. 1674–1681, 2018.
- [5] P. Maffezzoni, B. Bahr, Z. Zhang, and L. Daniel, "Oscillator array models for associative memory and pattern recognition," *IEEE Transactions on Circuits & Systems I Regular Papers*, vol. 62, no. 6, pp. 1591–1598, 2015.
- [6] A. Kumar and P. Mohanty, "Autoassociative memory and pattern recognition in micromechanical oscillator network," *Scientific Reports*, vol. 7, no. 1, pp. 411–419, 2017.
- [7] X. Liu, M. Kan, and W. U. Wanglong, "VIPLFaceNet: an open source deep face recognition SDK," *Frontiers of Computer Science*, vol. 11, no. 2, pp. 208–218, 2017.
- [8] N. Biyani, R. D. Righetto, and R. Mcleod, "Focus: the interface between data collection and data processing in cryo-EM," *Journal of Structural Biology*, vol. 198, no. 2, pp. 124–133, 2017.
- [9] Z. Zhou, H. Liao, B. Gu, K. M. S. Huq, S. Mumtaz, and J. Rodriguez, "Robust mobile crowd sensing: when deep learning meets edge computing," *IEEE Network*, vol. 32, no. 4, pp. 54–60, 2018.
- [10] L. Zhi, W. M. Wang, and L. Guo, "Toward open manufacturing: a cross-enterprises knowledge and services exchange framework based on blockchain and edge computing," *Industrial Management & Data Systems*, vol. 118, no. 9, pp. 303–320, 2018.
- [11] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5994–6009, 2017.
- [12] Z. Ke, Y. Mao, and S. Leng, "Mobile-edge computing for vehicular networks: a promising network paradigm with predictive off-loading," *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 36–44, 2017.
- [13] X. Lu, B. Qi, and H. Zhao, "A human Arm's mechanical impedance tuning method for improving the stability of haptic rendering," *Robotica*, vol. 39, no. 4, pp. 1–13, 2020.
- [14] L. N. Jessica, D. Z. Nancy, and V. S. Carlos, "Antimicrobial, optical and mechanical properties of chitosan–starch films with natural extracts," *International Journal of Molecular Sciences*, vol. 18, no. 5, pp. 1–18, 2017.
- [15] L. Resnik, H. Huang, A. Winslow, D. L. Crouch, F. Zhang, and N. Wolk, "Evaluation of EMG pattern recognition for upper limb prosthesis control: a case study in comparison with direct myoelectric control," *Journal of Neuro Engineering and Rehabilitation*, vol. 15, no. 1, pp. 1–13, 2018.
- [16] M. Papaioannou, E. Plum, and N. I. Zheludev, "All-optical pattern recognition and image processing on a metamaterial beam splitter," *ACS Photonics*, vol. 4, no. 2, pp. 217–222, 2017.
- [17] L. He, Y. Chen, Y. Wu, Y. Xu, Z. Zhang, and Z. Liu, "Nucleic acid sensing pattern recognition receptors in the development of colorectal cancer and colitis," *Cellular and Molecular Life Sciences*, vol. 74, no. 13, pp. 2395–2411, 2017.
- [18] Z. Lei, D. Qin, Y. Liu, Z. Peng, and L. Lu, "Dynamic energy management for a novel hybrid electric system based on driving pattern recognition," *Applied Mathematical Modelling*, vol. 45, no. 5, pp. 940–954, 2017.
- [19] S. Sahoo and M. K. Jha, "Pattern recognition in lithology classification: modeling using neural networks, self-organizing maps and genetic algorithms," *Hydrogeology Journal*, vol. 25, no. 2, pp. 311–330, 2017.
- [20] A. Goyal, H. J. Kwon, K. Lee et al., "Ultra-fast next generation human genome sequencing data processing using DRAGENTM bio-IT processor for precision medicine," *Open Journal of Genetics*, vol. 7, no. 1, pp. 9–19, 2017.
- [21] A. Zakia and M. A. R. Chaudhry, "A two-way street: green big data processing for a greener smart grid," *IEEE Systems Journal*, vol. 11, no. 2, pp. 784–795, 2017.
- [22] E. Pottier, W. M. Boerner, and S. R. Cloude, "Recent development of data processing in polarimetric and interferometric SAR," *URSI Radio Science Bulletin*, vol. 76, no. 1, pp. 48–58, 2017.
- [23] R. Schäfer, G. Schmidtke, T. Strahl, M. Pfeifer, and R. Brunner, "EUV data processing methods of the solar auto-calibrating EUV spectrometers (SolACES) aboard the International Space Station," *Advances in Space Research*, vol. 59, no. 9, pp. 2207–2228, 2017.
- [24] S. Li, Q. Cao, W. Xiao et al., "Optimization of acquisition and data-processing parameters for improved proteomic quantification by sequential window acquisition of all theoretical fragment ion mass spectrometry," *Journal of Proteome Research*, vol. 16, no. 2, pp. 738–747, 2017.
- [25] D. R. Kanniga, M. Gurusamy, and P. Vijayakumar, "An efficient cloud data center allocation to the source of requests," *Journal of Organizational and End User Computing (JOEUC)*, vol. 32, no. 3, pp. 23–36, 2020.
- [26] N. Zhang, R. Zhang, Z. Pang, X. Liu, and W. Zhao, "Mining express service innovation opportunity from online reviews," *Journal of Organizational and End User Computing (JOEUC)*, vol. 33, no. 6, pp. 1–15, 2021.
- [27] F. Meng, Q. Ji, H. Zheng, H. Wang, and D. Chu, "Modeling and solution algorithm for optimization integration of express terminal nodes with a joint distribution mode," *Journal of Organizational and End User Computing (JOEUC)*, vol. 33, no. 4, pp. 142–166, 2021.