Eng Int Syst (2021) 3: 167–173

Engineering
Intelligent Systems

Human Behavior Detection Based on RPN Network and Dynamic Image Recognition

Fei Yu* **and Zhaoxia Lu**

Shandong Sport University, Jinan, Shandong, 250000, China

As an important research direction in the field of computer vision and the core technology of artificial intelligence products, human behavior detection has received extensive attention from academia. Human behavior detection is the process of using computer vision technology to identify and locate pedestrians in images or videos. It has great research value and application potential in the fields of intelligent video surveillance, intelligent robots, and human-computer interaction, although issues of scale and viewing angle changes have always presented difficulties. Traditional human behavior detection methods yield unsatisfactory results when dealing with these two issues, and the detection speed is far from real-time requirements. This paper focuses on the three major detection difficulties: multi-scale, posture viewing angle change and real-time performance. The principle and simulation analysis of the classic algorithm DPM model in the traditional human behavior detection algorithm are carried out for the DPM model. In the feature extraction part, a fast feature pyramid model is proposed in order to improve the adaptability and real-time performance of pedestrian perspective and posture changes. An RPN network is introduced as a pre-detection module, and RPN + DPM and RPN+KCF detection models are proposed to improve small-scale detection capabilities and improve real - time processing capabilities. The simulation experiment is verified by means of VOC2007, MOT16 and other data sets. The detection performance and real-time performance of the proposed method have been significantly improved, and the neural network detection model shows obvious advantages.

Keywords: RPN; image recognition; gesture recognition; behavior detection

1. INTRODUCTION

This is an era of rapid scientific and technological development Computer technology and the Internet have gradually been integrated into many daily human activities such as study, work and leisure, and computers have become people's indispensable partners and assistants [1–3]. The rapid development of computer technology and the Internet has completely changed people's way of thinking and living [4].

Video behavior recognition technology, a sub-branch of the artificial intelligence domain, is generally a process in which a video is given and a computer is used to determine the kind of operation that the person or object of interest in the video is performing. Nowadays, because cameras are commonly used in all walks of life to record video data such as traffic management, family life and social events, live broadcast of various sports games, filming and television shooting, etc. video data has played an increasingly important role in most industries [5]. Therefore, in the information age, research based on computer vision has emerged, especially with the rapid development of science and technology in recent years and the substantial improvement of chip computing technology. Human behavior detection technology based on computer vision has attracted more and more attention from researchers. It has become an important area of research in the field of machine intelligence [6–8].

Computer vision-based behavior recognition technology has been studied by researchers for many years, with dynamic

*Address for correspondence: Fei Yu, Shandong Sport University, Jinan, Shandong, 250000, China, Email: yufei_lz@163.com

image recognition emerging as one aspect of image behavior recognition research [9]. Dynamic image recognition refers to a cropped image that contains only one action, and it requires people to use computer systems to automatically recognize and classify images. At present, the key difficulties of image behavior recognition technology are mainly the large environmental differences in the image content, the occlusion of the motion behavior in the image, the problem of the capture of multiple perspectives the illumination changes in the image, the low resolution of the image, and the complex dynamic image background and other factors. Moreover, with time, the changes in speed of the human body's actions in the image may also vary greatly, so it is often difficult to determine the starting point of the action in the image [10]. These factors also have a very large impact on the image extraction feature that represents the action. Therefore, the human behavior detection technology of dynamic image recognition has a long way to go before it can be widely promoted in various application fields [11].

2. RPN NETWORK AND THE THEORETICAL BASIS OF DYNAMIC IMAGE RECOGNITION

2.1 Faster RCNN Network

In the RPN network, in order to achieve the matching of the common feature frame in the original image with the candidate frame of each target, anchor boxes are used to set the target block diagram. It is known that each feature vector in the Conv feature map corresponds to a small area in the original image, and the size of the area is determined by the convolution kernel [12]. With each point on the feature map as the center, anchor boxes of different sizes and proportions can be generated to cover different areas on the original image. Originating from the actual changes in the size of the anchor boxes and the size of the target, usually this corresponding area cannot be completely matched with the target frame, resulting in the missing of the target in the RPN built-in candidate area [13]. In order to obtain a more accurate candidate area, it is necessary to optimize the preset area, and generate k frames of different proportions and different sizes to adapt to the target with each anchor point as the center to complete the area proposal link. Since the image sizes in the VOC data set are concentrated in the range of 500×375, the initial value of the design is able to conform to the distribution of the target size in the data set [14–16]. The generation parameters of the anchor frame are:

$$base\_size = 4 \tag{1}$$

$$ratio = [0.5, 1, 2] \tag{2}$$

$$scale = 1^{[3,4,5]} = [8, 16, 32] \tag{3}$$

After the RPN operation is completed, the output is a candidate frame containing only the foreground and position correction, and the Fast RCNN is used for classification and regression.

2.2 Gait Recognition and Dynamic Image Recognition Technology and Algorithms

The development process of this project mainly involves HTML5 technology, ASP.net technology framework, gait information generation technology, dynamic image recognition technology and deep neural network.

2.2.1 Gait Recognition Technology and Dynamic Image Recognition Technology

In the current era of information technology, biometric technology has gradually become a research focus. Biometrics is an identification method based on the physiological or physiological characteristics of the human body. Its aim is to digitize various personal biological information and extract the identifiable features for comparison, thereby realizing biometrics [17]. Various aspects of the current biometric technology have been successfully applied For example, human fingerprints, iris, face, palm prints, handwritten fonts, voices, etc. can all be used for biometric recognition. The application of this information is characterized by relatively high accuracy. However, the collection method is more complicated. Relatively speaking, the gait of a person walking can also become a biometric feature. Psychology, medicine and biomechanics experiments have proved that everyone has his/her own characteristic gait pattern. Albeit slightly inaccurate, the use of gait to derive information about a subject has its advantages: (1) gait recognition is non-invasive and can be recognized without the need for contact; (2) image quality at low resolution can also detect and measure gait while maintaining high recognition accuracy; (3) it can be applied to special environments - infrared measurement can be used to perceive gait at night; (4) because walking is a subconscious human behavior it is difficult to imitate or tamper with the gait feature

Identification and classification methods can be either model-based or model-free, depending on the types of features used. Model-based methods generally use measurable and quantifiable kinematic parameters as features [18]. For example, in the human body parameter method, there are more methods that use the distance between the human head and the feet, the distance between the head and the pelvis, and the feet. The distance between the body and the pelvis and the distance between the left foot and the right foot are used as parameters to form two sets of static body and stride parameters. This method has nothing to do with the measurement scale and performs well under the same viewing angle, but it is very sensitive to the quality of the gait sequence. At the same time, the calculation cost is relatively high due to the need for a large number of parameter calculations [19]. On the other hand, model-free methods generally consider the problem of gait recognition as a whole, and do not pay attention to human body structure and parameters. For example, one method is to divide the image into several overlapping parts consisting of the upper part, the middle part and the lower part, and the left and right parts, and then train a Bayesian network for recognition. Compared with the model-based method, the model-free method has faster recognition speed, but it is more sensitive to changes in external conditions such as changes of viewing angle and illumination.

2.2.2 Gait Recognition Process and Algorithm

Each background image needs to be processed to establish a background model corresponding to each camera. In this project, the establishment of the background model adopts the minimum median square method, which is characterized by the ability to use multiple background images to obtain a relatively accurate background model. From a theoretical perspective, the background model can be regarded as the gray value of each pixel, denoted as $gv_{x,y}$, where x, y are the corresponding abscissa and ordinate values in the image. Suppose there are k background images, denoted as image $(1 \leqslant i \leqslant k)$, then $gv_{x,y}$ need to meet the following conditions:

$$Min\left(\sum_{i=1}^{k}(image\_gv(x,y)^i - gv_{(x,y)})^2\right) \tag{4}$$

After the background model is obtained, the contour of the pedestrian can be obtained by comparing the image with the background model, and the difference operation of the image pixel or gray value is enough. In this project, the pedestrian contour difference operation function is selected, which is defined as follows:

$$Min\left(\sum_{i=1}^{k}\left(image\_gv_{x,y}^{i} - gv_{x,y}\right)^2\right)$$
$$diff_{x,y}(image) = 1 - \frac{2\sqrt{(image\_gv_{x,y}+1)(gv_{x,y}+1)}}{(image\_gv_{x,y}+1)+(gv_{x,y}+1)}$$
$$- \frac{2\sqrt{(256-image\_gv_{x,y})(256-gv_{x,y})}}{(256-image\_gv_{x,y})+(256-gv_{x,y})} \tag{5}$$

where image is the image input by the camera, $gv_{x,y}$ are the gray values of the background model. For a preset threshold δ, a binary differential image DI can be generated, and the gray value formula of the pixel point is:

$$DI_{x,y} = \begin{cases} 1, & if \quad diff_{x,y}(image) \geq \delta \\ 0, & else \end{cases} \tag{6}$$

3. CONSTRUCTION OF HUMAN BEHAVIOR DETECTION SYSTEM AND ANALYSIS OF TEST RESULTS

3.1 Demand Analysis of Human Behavior Detection Platform

In this study, a human behavior detection platform is established in order to achieve the main research goal: to use deep learning algorithms to recognize and detect the behavior of three-dimensional video streams in the monitoring platform; to detect abnormal behaviors through real-time monitoring and alert relevant personnel, and develop a real-time online monitoring platform with a high rate of recognition accuracy. Compared with the traditional video surveillance platform, the intelligent surveillance platform has become a new application direction in the artificial intelligence industry because of its intelligent alarm function. Its intelligence lies in the use of cameras to replace human eyes and deep learning networks to replace human brains to analyze the video stream in the video surveillance platform and give the result. Because the platform is automatic, it can analyse abnormal events without the need for human intervention and promptly alert relevant personnel about the need to deal with a situation, thereby reducing human resources to a certain extent and alleviating the burden on staff.

Human behavior detection platforms require the following:

1. Development of an abnormal behavior recognition algorithm with strong universality, improve the robustness of the algorithm, and make it adapt to different behaviors and scenarios as much as possible.

2. Development of an intelligent behavior analysis platform based on the current monitoring platform to undertake the analysis of abnormal events and provide real-time alarms.

3. Development of a complete set of intelligent monitoring platforms, including desktop applications that interact with customers, a recognition server for behavior analysis, and a database server for the purpose of archiving and viewing abnormal behavior data.

3.2 Human Behavior Detection System Structure

3.2.1 Platform Architecture Design

This platform adopts a highly functional C/S architecture. The overall architecture of the platform is shown in Figure 1 below. This platform consists of a deep learning platform, business application layer and interface layer. The deep learning platform includes resources, scheduling, storage, and algorithms. Due to the large amount of calculations required, deep learning needs a GPU with strong computing power as its hardware foundation.

3.2.2 Platform Workflow

The workflow of the entire human behavior detection platform is shown in Figure 2. First, start the database server and initialize it to wait for the client's connection access. Second, the client application is opened, the user logs in and connects to the server, and initializes the client. The initialization includes the configuration information of the camera and the initialization of the user interface. The camera information on the client can also be added to or modified, and displayed on the multi-screen interface. Then, select the camera of interest and submit it to the server. The server starts the corresponding thread in the background according to the camera information submitted, and sends the result of the thread processing to the client in real time through the intelligent analysis server, and the client receives the alarm information The relevant personnel will be able to deal with abnormal events in a timely manner through voice or text messages. Moreover, the server has an exception handling module. No matter which part of

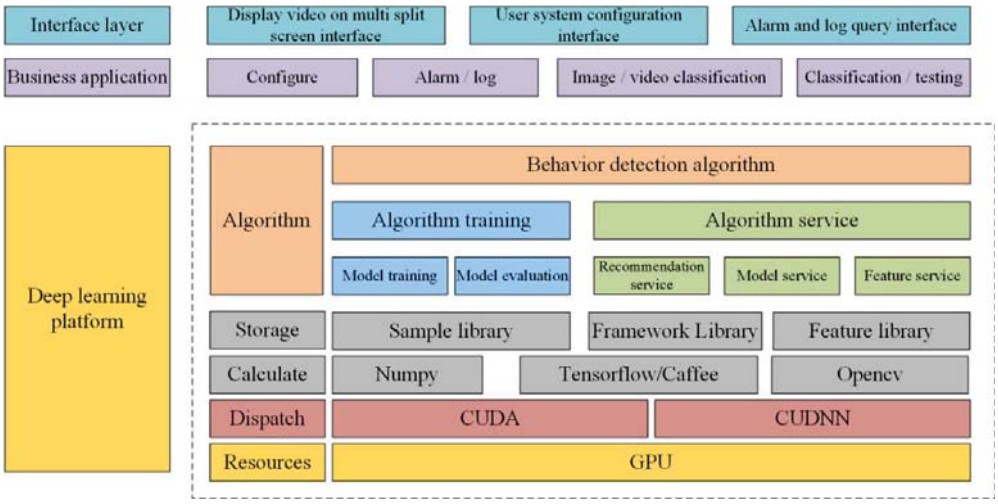

Figure 1 Human behavior detection platform architecture.

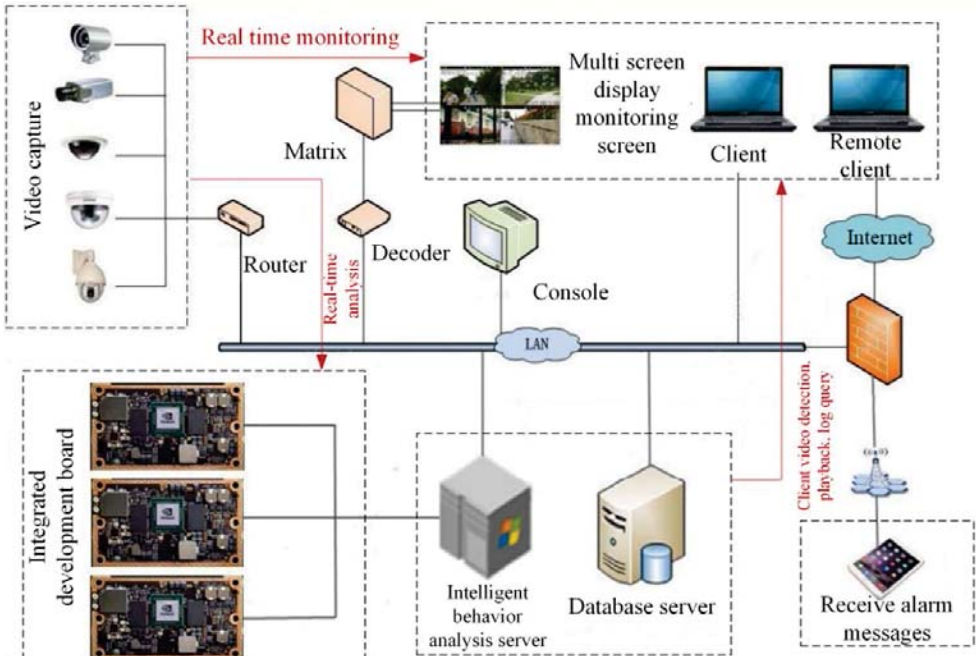

Figure 2 The workflow of the human behavior detection platform.

the platform has a problem, the exception handling module will immediately raise the alarm and issue an exception announcement for the client. The abnormal information is saved on the log database, waiting for maintenance personnel to deal with the situation

3.3 Platform Development and Operating Environment

Both the intelligent behavior analysis server system and the database server system are developed on the Ubuntu 16.04 system. The intelligent behavior analysis server system uses python language as the development language, and the database server system uses C++ language as the development language. Abnormal behavior detection uses the basic opensource platforms TensorFlow and darknet for deep learning, and uses the Opencv computer vision library to perform basic preprocessing operations on images; the client system is developed on the Windows system, with C++ as the development language The operation of the background detection algorithm is undertaken by the embedded development board, and is called by the intelligent analysis server. The development environment of the human behavior detection platform is shown in Table 1.

3.4 Platform Real-Time Optimization and Test Analysis

3.4.1 Camera Video Decoding Optimization

The camera video stream is obtained through OpenCV decoding, but due to its slow decoding speed, there will be

Table 1 Development environment of human behavior detection platform.

	Surroundings	Behavior analysis server	Database server	Client
Software configuration	Operating platform	Ubuntu16.04 64bit	Ubuntu16.04 64bit	Windows 8 64bit
	Integrated development environment	Anaconda2	QT 6.5	QT 6.5
Hardware environment	CPU	Intel core i7-4790 @4.00GHz×8	Intel core i7-7700 @3.60GHz 16G	Intel(R) Core i7-8550 @1.80GHz 8G
	GPU	TITAN×(pAscal) /PCIe/SSE2 12GiB	GeForce GTX 970 8GiB	Intel(R) UHD Graphics 620
	RAM	16GiB	8GiB	8GiB
Other	Deep learning basic open-source platform	Tensorflow, Darknet, Caffe	–	–
	Computer vision library	Opencv3.2.0 PIL	–	–
	Development language	Python2.7.1	C++	C++
	Video decoding library	Haikang open-source video decoding library	–	Haikang open-source video decoding library

Table 2 Comparison of camera video decoding speed.

Camera resolution	CPU decoding(s)	GPU decoding(s)	SDK decoding interface(s)
1080p(1s)	0.082	0.052	0.045
720p(1s)	0.061	0.036	0.025

Table 3 Platform accuracy test results.

Detection environment	Violence detection	Fall detection	Wandering detection	Intrusion detection
Indoor environment	95%	98%	100%	100%
Outdoor environment	91%	92%	95%	99%

frame loss. In order to improve the detection speed on the server side and the client side to enable the simultaneous playback of multiple cameras, we call the device network SDK decoding interface provided by Haikang to analyze the video stream of the webcam. The video stream of the camera is compared through the GPU accelerated decoding method and the Haikang SDK decoding interface is called. It can be seen that the decoding speed of the camera has been greatly improved through the optimization of the GPU acceleration and the SDK decoding interface. The results are shown in Table 2 below.

3.4.2 Platform Testing and Analysis

The accuracy of this platform is tested in two environments: indoor and outdoor. The abnormal behaviors detected by each camera are counted. Undetected abnormal behaviors include false positives and false negatives. Through the test, we know that although violence is not as high as other behaviors, its detection accuracy has reached our expected goal. The test results are shown in Table 3.

This platform supports real-time detection by multiple cameras. Embedded platforms and desktop computers have three and four separate cameras. During the test, the real-time performance of the platform was tested by calculating the time required for the network to analyze one minute of video and calculating the average time required for each camera. The test results are shown in Table 4 below.

From the real-time test results of the platform shown in the table above, it can be seen that as the number of channels increases, the time-consuming platform detection will be longer, which means that the detection speed will decrease. However, the total time consumption does not affect the real-time requirements of our platform.

4. CONCLUSION

Pedestrian detection technology is a hot topic as it presents difficult problems in regard to computer vision, particularly as it has a wide range of applications in robotic navigation, image retrieval, automatic driving, and intelligent video surveillance [20–22]. With the development of RPN technology and the maturity of computer hardware platforms, pedestrian detection algorithms based on dynamic image recognition have received more attention. Compared with traditional pedestrian detection algorithms, those based on dynamic image recognition are faster and have greater accuracy and precision [23]. They perform better and, moreover, yield better results where complex scenes are involved. The paper focuses on the three difficulties facing human behavior detection: the multi-scale issue, posture viewing angle

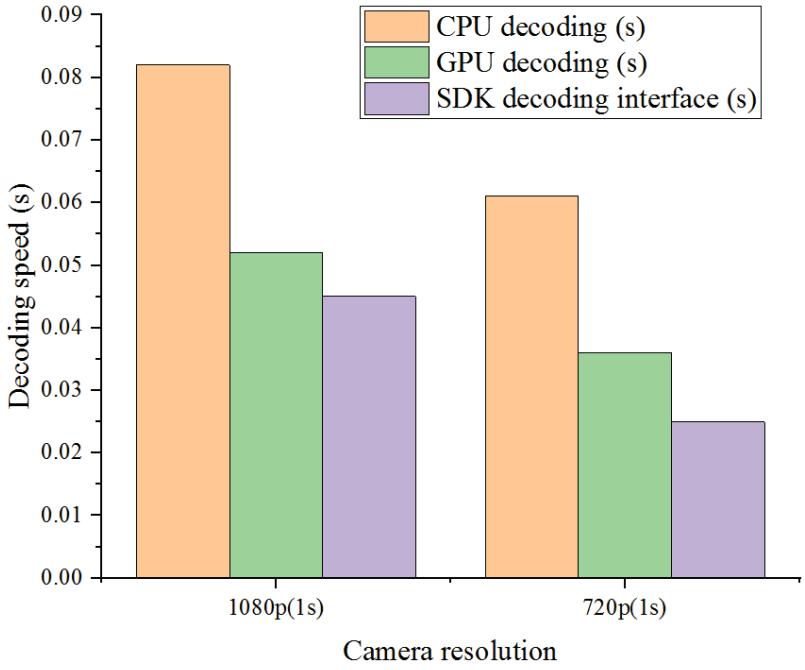

Figure 3 Comparison of camera video decoding speed.

Table 4 Platform real-time test results.

Number of cameras recognized in parallel	Time to detect one minute of video(s)	Average time per camera(s)
1	6.3	3.30
3	6.82	3.19
5	7.51	3.50
8	7.81	3.08
10	8.28	3.23
13	9.19	3.22
15	9.86	3.28
18	10.95	3.36
21	12.16	3.60

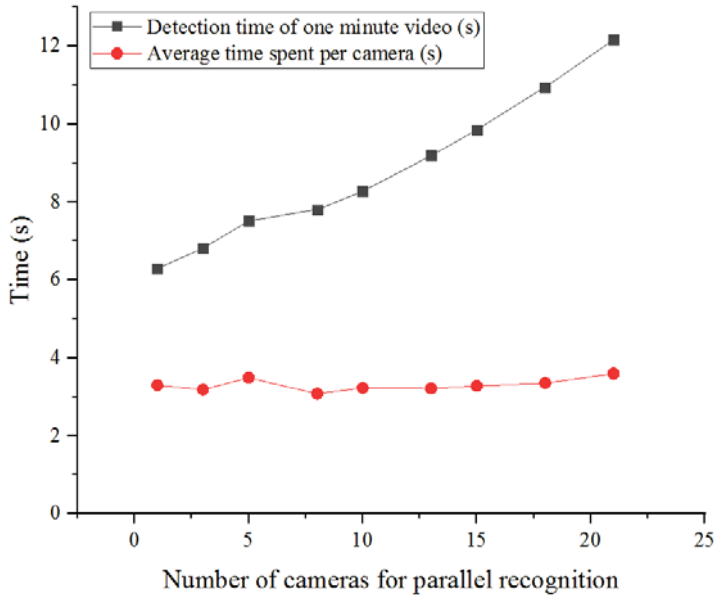

Figure 4 Platform real-time test results.

change, and detection in real time. The adopted method has made significant progress in the detection of small-scale targets and targets with large viewing angle changes and detection speed [24]. It is significantly improved, and a pedestrian detection method based on RPN network and related filtering is proposed. Due to its excellent video representation ability, a better recognition and classification result is obtained, which provides a certain reference for human behavior detection in dynamic image recognition.

REFERENCES

1. AA. Elsadek and BE. Wells A heuristic model for task allocation in heterogeneous distributed computing system. Int J Comput Appl 6(1) (1999), 0–35.
2. HR. Faragardi, R. Shojaee, MA. Keshtkar, H. Tabani Optimal task allocation for maximizing reliability in distributed real-time systems. In: IEEE/ACIS 12th international conference on computer and information science (ICCIS) 24(3) (2013), 746–758.
3. DK. Govil and DA. Kumar A modified and efficient algorithm for static task assignment in distributed processing environment. Int J Comput Appl 23(8) (2011), 1–5.
4. R. Gupta and PK. Yadav Task allocation model for balance utilization of available resource in multiprocessor environment. J Comput Eng 17(4) (2015), 94–99.
5. CC. Hsieh and YC. Hsieh Reliability and cost optimization in distributed computing systems. Comput Oper Res 30(8) (2003), 1103–1119.
6. M. Kafil and I. Ahmad Optimal task assignment in heterogeneous distributed computing systems. Complex Distrib Syst 6(3) (1998), 42–50.
7. S. Kalyani and KS. Swarup Supervised fuzzy c-means clustering technique for security assessment and classification in power systems. Int J Eng Sci Technol 2(3) (2010), 175–185.
8. Q-M. Kang, H. He, H-M. Song, R. Deng Task allocation for maximizing reliability of distributed computing systems using honeybee mating optimization. J Syst Softw 83(1) (2010), 2165–2174.
9. S. Kartik and CSR. Murthy Task allocation for maximizing reliability of distributed computing systems. IEEE Trans Comput 46(6) (1997), 719–724.
10. U. Kaushal and A. Kumar Improving the performance of DRTS by optimal allocation of multiple tasks under dynamic load sharing scheme. Int J Sci Eng Res 4(7) (2013), 1316–1321.
11. Y. Kopiddakis, M. Lamari, V. Zissimopoulos On the task assignment problem: two new heuristic algorithms. J Parallel Distrib Comput 42(1) (1997), 21–29.
12. PR. Kumar and S. Palani A dynamic voltage scaling with single power supply and varying speed factor for multiprocessor system using genetic algorithm. In: Proceedings of the international conference on pattern recognition. Informatics and medical engineering 54(3) (2012), 342–346.
13. H. Kumar and I. Tyagi Implementation and comparative analysis of k-means and fuzzy c-means clustering algorithms for tasks allocation in distributed real time system. Int J Embed Real Time Commun Syst 10(4) (2019), 66–86.
14. A. Kumar and PK. Yadav Task management algorithm for distributed system. In: 15th International conference of international academy of physical science 59(3) (2012), 825–831.
15. A. Kumar, V. Upadhyay, SK. Dubey Optimal approach for tasks allocation based on fusion of unallocated tasks in distributed systems. Int J Adv Res Comput Sci 8(3) (2017), 602–607.
16. H. Kumar, NK. Chauhan, PK. Yadav A high performance model for task allocation in distributed computing system using k-means clustering technique. Int J Distrib Syst Technol 9(3) (2018) 1–22.
17. H. Kumar, NK. Chauhan, PK. Yadav Hybrid genetic algorithm for task scheduling in distributed real-time system. Int J Syst Control Commun 10(1) (2019), 32–52.
18. CH. Lee, D. Lee, M. Kim Optimal task assignment in linear array networks. IEEE Transact Comput 41(2) (1992), 877–880.
19. W. Li, FC. Delicato, PF. Pires, et al., Efficient allocation of resources in multiple heterogeneous wireless sensor networks. J Parallel Distrib Comput 86(3) (2014), 76–81.
20. VM. Lo Heuristic algorithms for task assignment in distributed system. IEEE Trans Comput 37(11) (1988), 1384–1397.
21. YC. Ma, TF. Chen, CP. Chung Branch-and-bound task allocation with task clustering-based pruning. J Parallel Distrib Comput 64(5) (2004), 1223–1240.
22. A. Mahmood Task allocation algorithms for maximizing reliability of heterogeneous distributed computing systems. Control Cybern 30(1) (2001), 115–130.
23. RA. Na'mnch and KA. Darabkh A new genetic-based algorithm for scheduling static tasks in homogeneous parallel systems. In: International conference on robotics, biomimetics, intelligent computational systems (ROBIONETICS) 59(6) (2013), 46–50.
24. M. Naderam, M. Dehgham, S. Goddard Upper and lower bounds for dynamic cluster assignment for multi-agent tracking in heterogeneous WSNs. J Parallel Distrib Comput 73(10) (2012), 1389–1399.