BILGISAYAR BILIMLERI VE TEKNOLOJILERI DERGISI
BIBTED
MERSIN UNIVERSITESI 1962

Araştırma Makalesi

A Distributed Computer System Architecture Acting As A High Performance Computing System For Big Data Analytics

Hamza EROL*[1], Timuçin KORKMAZ[2]

[1]*Mersin University, Engineering Faculty, Department of Computer Engineering, Mersin - Turkey*
[2]*Mersin University, Graduate School of Natural and Applied Sciences, Department of Computer Engineering, Mersin - Turkey*

Keywords:
Big data analytics, Distributed computer system architecture, Emotion analysis, Image classification, Parallel programing, High performance computing system, Face expression recognition

ABSTRACT

A distributed computer system architecture, which acts as a high-performance computing system for big data analytics, was created. In the distributed computer system architecture, the volume of big data and the load of the threads to be processed were taken into account. The distributed computer system architecture was designed based on the principle of using personal computers together and on the same computer network in normal standards and features. For the distributed computer system architecture, a computer featured on the common computer network is planned as the main computer and the others as slave computers. Threads created for big data analytics were processed simultaneously in the processors of both the main computer and slave computers. Two different databases were used in the processing of threads created for big data analytics in the distributed computer system architecture. A database was used for the process of reading data files for threads and another database was used for writing process results to output files. Visual C # parallel programming language was used to process the threads in the processors of computers in the distributed computer system architecture. In this study, the emotional expression classification of facial images was made. For this purpose, first the image processing was done in the MATLAB software, and the creation of threads was carried out. The threads created later were processed simultaneously on the processors of computers in the distributed computer system network. By processing the threads in the distributed computer system architecture, image files are classified for facial emotion prediction. Process outputs and file classification labels for threads are printed as output to the second database. Big data analytics can be done using high performance computing systems. Such systems are costly, and the number of installed systems is low. With this study, the distributed computer system architecture was created for big data analytics. High performance computing system architecture was realized by using databases and parallel programming. The image files in the database are classified and processed as threads in the high-performance computing system architecture created. The cost of the created system is low, and the speed of processing is high. With the distributed computer system architecture, big data analytics has been made convenient and applicable.

*Sorumlu Yazar

*(herol@mersin.edu.tr) ORCID ID 0000 – 0001 – 8983 – 4797
(timucinkorkmaz33@gmail.com) ORCID ID 0000 – 0003 – 0888 – 1327

e-ISSN: 2717-8579
Geliş Tarihi: 06/10/2020; Kabul Tarihi:13/12/2020l

Büyük Veri Analitiği İçin Yüksek Performans Hesaplama Sistemi Gibi Davranan Bir Dağıtık Bilgisayar Sistemi Mimarisi

Anahtar Kelimeler:
Büyük veri analitiği,
Dağıtık bilgisayar sistemi mimarisi,
Duygu analizi,
Görüntü sınıflandırma,
Paralel programlama,
Yüksek performans hesaplama sistemi,
Yüz ifadesi tanıma

ÖZET

Büyük veri analitiği için yüksek performans hesaplama sistemi gibi davranan bir dağıtık bilgisayar sistemi mimarisi oluşturulmuştur. Dağıtık bilgisayar sistemi mimarisinde, büyük verinin hacmi ve işlenecek iş parçacığının yükü dikkate alınmaktadır. Dağıtık bilgisayar sistemi mimarisi, kişisel bilgisayarların birlikte ve aynı bilgisayar ağı üzerinde normal standartlarda ve özelliklerde kullanılması esasına göre tasarlanmıştır. Dağıtık bilgisayar sistemi mimarisi için ortak bilgisayar ağında yer alan bir bilgisayar ana bilgisayar, diğerleri de bağımlı bilgisayar olarak planlanmıştır. Büyük veri analizi için oluşturulan iş parçacıkları hem ana bilgisayar hem de bağımlı bilgisayarların işlemcilerinde aynı anda işlenmektedir. Dağıtık bilgisayar sistemi mimarisinde büyük veri analitiği için oluşturulan iş parçacıklarının işlenmesinde iki farklı veri tabanı kullanılmaktadır. İş parçacıkları için veri dosyalarını okuma işlemi için bir veri tabanı ve işlem sonuçlarını çıktı dosyalarına yazmak için başka bir veri tabanı kullanılmaktadır. Visual C# paralel programlama dili, dağıtılmış bilgisayar sistemi mimarisindeki bilgisayar işlemcilerindeki iş parçacıkları işlemek için kullanılmıştır. Bu çalışmada yüz görüntülerinin duygusal ifade sınıflandırması yapılmıştır. Bu amaçla önce MATLAB yazılımında görüntü işleme yapılmış ve internet protokol numaralarının oluşturulması gerçekleştirilmiştir. Daha sonra oluşturulan iş parçacıkları, dağıtılmış bilgisayar sistemi ağındaki bilgisayarların işlemcilerinde eşzamanlı olarak işlenmiştir. Dağıtılmış bilgisayar sistemi mimarisinde iş parçacıkları işlenerek, görüntü dosyaları yüz ifadesi tahmini için sınıflandırılmıştır. İşlem çıktıları ve iş parçacıkları için dosya sınıflandırma etiketleri, ikinci veri tabanına çıktı olarak yazdırılmıştır. Büyük veri analizi, yüksek performanslı bilgi işlem sistemleri kullanılarak yapılabilmektedir. Bu tür sistemler maliyetlidir ve kurulu sistem sayısı düşüktür. Bu çalışma ile büyük veri analitiği için dağıtık bilgisayar sistemi mimarisi oluşturulmuştur. Veri tabanları ve paralel programlama kullanılarak yüksek performanslı bilgi işlem sistemi mimarisi gerçekleştirilmiştir. Veri tabanındaki görüntü dosyaları, oluşturulan yüksek performanslı bilgi işlem sistemi mimarisinde iş parçacıkları olarak sınıflandırılır ve işlenir. Oluşturulan sistemin maliyeti düşük ve işlem hızı yüksektir. Dağıtılmış bilgisayar sistemi mimarisi ile büyük veri analitiği rahat ve uygulanabilir hale getirilmiştir.

*Sorumlu Yazar

*(herol@mersin.edu.tr) ORCID ID 0000 – 0001 – 8983 – 4797
(timucinkorkmaz33@gmail.com) ORCID ID 0000 – 0003 – 0888 – 1327

e-ISSN: 2717-8579

Geliş Tarihi: 06/10/2020; Kabul Tarihi:13/12/2020l

1. INRODUCTION

In this study, the distributed computer system architecture was created by installing an intranet among the required number of personal computers. It was made by pinging on the manual IP route, which is connected to each other with personal computers. While IP is given to personal computers, network and internet settings are regulated. Network and internet settings are started. Adapter options are changed. Ethernet card is set. The properties of the Ethernet card are regulated. Internet protocol version 4 (TCP / IPv4) features are regulated. IP values are entered.

By entering the IP address when entering IP values, the connection is provided by entering the desired IP and subnet mask values. Connection control is carried out with the ping command via cmd. While making transactions, IP addresses based on 192.168.0.1 have been preferred to computers. In this study, the IP address of the main computer (PC1_M00) was determined as 192.168.0.14 and the IP addresses of the slave computers (PC2_Sxx) was determined as the number required, starting from 192.168.0.15. Slave operations are required to access the main computer, as the operations are done on the main computer. Access test was done by typing ping 192.168.0.14 code on the cmd screen of the slave computers.

Pinging from the slave computers (PC2_Sxx) ensures that access to the main computer (PC1_M00) has been successful. The connection between the number of personal computers required in the basic distributed computer system architecture is an intranet connection. The distributed computer system architecture also has switches with the required number of ports for the common computer network. If desired, parallel switches can be made with the processor of more computers by increasing the number of slave computers by placing switches in the distributed computer system architecture.

In this study, a distributed computer system architecture (Yadav et al., 2013), which acts as a high performance calculation system (Cliff et al., 2019), was created for big data analytics (Ezhilraman and Srinivasan, 2018; Kouanou et al., 2018). In this section, big data analytics, high performance calculation system, distributed computer system architecture (Czarnul et al., 2020), parallel programming (Hager and Wellein, 2011), database programming, computer architecture (Murdocca and Heuring, 1999), computer network (Dordal, 2020) concepts will be examined in relation.

Distributed computer system architecture, created to classify image files in the database and acting as a high-performance calculation system, will be explained with the concepts of database and parallel programming.

2. DISTRIBUTED COMPUTER SYSTEM ARCHITECTURE

2.1. System and Application Softwares

A computer network system is called a distributed computer network in the system where computers communicate with each other using the network infrastructure of hardware or software tools (Coulouris et al., 2012). A distributed computer system can have two or more computers and/or devices. These devices can make data exchange with each other over the computer network. In the distributed computer system architecture, MSSQL database has been installed on the main computer. In the distributed computer system architecture designed in this study, all personal computers were connected to the MSSQL database installed on the main computer. Information about the operations performed in the classification of face images and the transaction outputs are printed on the database. In addition, database programming has been checked to see if the image is processed in the database. MSSQL Management v17.7 version was installed and used on the main computer.

In the distributed computer system architecture, MSSQL database server permissions were established by opening 1433 port over the network so that all personal computers can access the MSSQL database server. This port must be opened in order to access the MSSQL database server over the network.

The process of opening ports on the MSSQL database server are: (1) SQL Server Configuration Manager opens. SQL Server Configuration Manager can be accessed via the path C:\Windows\SysWOW64\SQLServerManager14.msc. In this screen, SQL Server Network Configuration is selected. (2) The TCP/IP setting is set to Enabled on the incoming screen. (3) Right click TCP/IP option and select features. The IP Addresses tab is displayed on the screen. (4) By finding the IP address to be used, the Enabled field is set to Yes and the port information to be forwarded is entered. (5) After all these operations are done, the screen is closed by saying Apply changes. Restart is done by selecting SQL Server Services for the changes to be applied. (6) After opening the port, the permission process for this port is required from the Firewall. Permission is made so that another device can access it. For this, Control Panel->System and Security->Firewall screen opens. Advanced settings are selected on the selected screen. (7) Select Rules from Screen, Click New Rule button. (8) Select the port on the screen and press the next button. (9). On the incoming screen, TCP/IP is selected, as a rule, by selecting specific local ports, enter the port information to be allowed here and click on the next option. (10). Select the Allow Connection button on the Action screen and click Next. (11) Marked in 3 options on the

Profile screen and called Next. (12) Finally, the Last Option is selected by naming the rule. (13) The Rule allowed after this Transaction will appear on the rules screen. Access to the SQL Server is permitted via the IP address allowed. (14) You can enter the system by calling Connect. When it is entered into the system by connecting to it, it writes the IP address determined as the name of SQL Server in the section marked in red. This SQL Server can be accessed through the same network infrastructure.

Another software used for distributed computer system architecture is the Entity framework, a Microsoft-based framework. Entity framework is a process that enables data to be sent to the database in object oriented programming with a relational database (Lerman, 2009). The necessary transactions are provided by transmitting the data to the database through the framework. Relational database operations with the Entity framework are performed using the model structure.

Entity Framework is based on Microsoft. The ADO.NET (ActiveX Data Objects.NET) infrastructure is used when making data connections. ADO.NET is a Microsoft-based data access and management technology. Entity Framework is a more advanced structure than ADO.NET technology. It is necessary to use query clauses for all database operations in ADO.NET. Performing a query requires more processing than current technologies. All database operations in Entity framework; is done on the created model. Entity Framework uses 3 different methods for 4 different situations in the workflow (Hung et al., 2016). There are advantages and disadvantages to using the Entity Framework. The advantages of using Entity Framework are: (1) It provides object-oriented project development. (2) Allows to perform transactions on the database without database knowledge. (3) Used with all database applications. (4) No need to back up the database with Code first technique. Operation can be done by creating a database at the desired location. (5) It positively contributes to software development time and costs.

The disadvantages of using Entity Framework are: (1) Processing speed is slower than ADO.NET. (2) All control operations are carried out by the entity framework. This sometimes creates complex queries in an easy check and negatively affects performance. (3) Linq is used as the language of use.

2.2. Databases in Basic Distributed Computer System Architecture

First, Information on two different databases used in the distributed computer system architecture created in this subsection will be given. The first database is the MSSQL database used by the distributed computer system architecture. The second database is the MS Access database used in the classification process. The first database used in the distributed computer system architecture is SQL based. In the basic distributed computer system architecture, this database accessed by computers is the database containing information about the data analyzed in the system.

In the database, there is an automatically increasing Id related to the image, the name of the image being processed, the emotional state detected in the image, the computer information processing and the date of the transaction. A database named GoruntuAktar was created in the database by entering the appropriate formats for the state of the data. The created database table and the format information of the data are kept in the DataAktar table.

In this study, while the application developed with visual C# parallel programming is running and before the application analysis result is saved, the registration process is performed by recording the data status in this table about the related image. If there is no information about the image to be analyzed in the related table, the analysis process begins. If there is information about the image to be analyzed in the related table, the application continues to operate with the other image. The developed application process makes another final check before saving the result to the database at the end of the analysis. This is because the image may have been processed and recorded in the same time period on another computer's processor. Particular attention was paid to the absence of duplicate records for the analysis results to be correct. As a result of the recheck, if the result related to the image information is not in the database, the result of the analysis is recorded in the database. Application software developed with visual C# parallel programming saves the analysis results to the table in the database after data processing, data labeling and classification.

MS Access database is the second database used in data analysis in the distributed computer system architecture. MS Access database has been preferred because the analysis process can be done in a short time and because there is not a lot of processing load. By recording an image about each emotion in the MS Access database, the analysis process is performed with similarity to these images. The database created contains right eye, left eye position information, mouth opening rate, and emotional state corresponding to this image. In the data analysis application, the analysis process of the image is completed and the emotion value closest to the values found is calculated and the result of the operation is determined according to the emotion in that image. Right eye, left eye position information, mouth opening rate, and mood values corresponding to this image are added to the created MS Access database. In the database used, a test data table was created containing different emotions from different data sets. According to the results obtained, emotion results are obtained by making a comparison with

the data table. While obtaining the results, transactions are made according to the values in the data table.

With some images that are recorded in the system and provide the formation of data sets, training was made according to the different emotional states of the same person. There are differences in facial expressions according to the images on the faces. Using different faces with the same emotion enables the data to be analyzed correctly. The fact that the facial data used is different and diverse increases the accuracy and reliability of the results obtained. Different emotions of the same person in the image files recorded in the database are shown in Figure 1.

Figure 1. Different emotions of the same person in the image files stored in the database.

The application software developed with visual C# parallel programming determines the emotional values corresponding to the image according to five different facial emotions using the right eye, left eye position information, mouth opening ratio values in the created MS Access database. In the MS Access database created by the application software, using the values of right eye, left eye position, mouth opening ratio, the mood values corresponding to the image were obtained according to five different facial emotions and added to the table. Processing is performed according to the id data of the line containing the results of the closest values. The obtained Id gives the mood information of the person.

3. THE PROPOSED METHOD

The distributed computer system (Czarnul et al., 2020) proposed in this study consists of the required number of normal standard personal computers. The computer, which has better features than other computers, defined as the main computer (PC1_M00) and others are designated as slave computers (PC2_Sxx). All computers defined and introduced to each other in the computer network. General features of personal computers used in the distributed computer system architecture: Main computer (PC1_M00): Processor: Intel Core I7-4720HQ CPU 2.60 GHz; RAM Memory: 12 GB; System Type: 64 bits. Slave computers (PC2_Sxx): Processor: Intel Core I5-3210M CPU 2.50 GHz; RAM Memory: 6 GB; System Type: 64 bits. The personal computers are connected to each other via an Ethernet cable via CAT5 / CAT6 local area network port.

In the architecture of the distributed computer system created in this section, the operations performed in parallel programming phase for big data analytics will be explained.

3.1. Threads and Parallel Programming For Image Files In The Database

In the distributed computer system architecture, in order to perform big data analytics with database and parallel programming, threads are taken in parallel programming stage as much as the number of image files in the database. The data of the received image files are sent for processing in the processors on the computer network and the analysis process is started. System allows to keep information about which thread is processed with which file with which line of code:

FileVM dd = new FileVM {ThreadID = Thread.CurrentThread.ManagedThreadId, ThreadName = folder};

The process, which completes the thread, comes to the selection stage and takes an image file again and the process takes place. The image files are taken as threads and processing steps are performed in parallel programming. The content and information of the image received as a thread is sent to a computer on the common network in the distributed computer system architecture for processing in the processor. The properties of the image taken as a thread can also be displayed. The features of the images in the database include information such as name, width, length. While the image file is processed in the processor, the software classification developed with visual C# parallel programming is performed according to the image's width and height information. After the face shape determination process, the edge and the edge of the eyes are detected by performing pixel processing in the width and length of the image.

3.2. Processing of Image Files in The Database with Parallel Programming

In the application developed using visual C# parallel programming, MATLAB software was first used for processing image files. In practice, it first separates an image sent to the processors in the distributed computer system architecture as pixels in RGB. The face in the image is emphasized by performing a 3% cut/trim on each edge of the image. With the imcrop function of the MATLAB program, cutting/trimming is performed from the edges of the image. After determining the face shape from the image, the edge and eye edge detection process is performed with pixel operations from the width and length of the image. With the developed application, the edges of the images to be processed in the database are cropped. Facial elements in the cropped image are more easily determined. This facilitates the detection event of edge detection filters.

3.3. Processing of Image Files in The Database with Parallel Programming

Within the scope of big data analytics, using distributed computer system architecture, detailed face edge detection operations were performed in image files in the database (Acharjya et al., 2012; Shrivakshan and Chandrasekar, 2012). Edge detection filters have been applied to image files in the database. The 4 different edge detection methods used in the study are: (1) Canny edge detection. (2) Laplace Edge Detection. (3) Sobel Edge Detection and (4) Robert Edge Detection.

Four different edge detection algorithms have been applied to face image files in the database for the face recognition system (Tsankashvili, 2018). Accordingly, all face image files in the database are classified into five different emotion categories. The results are given in Table 1.

TABLE 1. CLASSIFICATION OF ALL FACE IMAGE FILES IN THE DATABASE INTO FIVE DIFFERENT EMOTION CATEGORIES USING FOUR DIFFERENT EDGE DETECTION ALGORITHMS FOR THE FACE RECOGNITION SYSTEM.

		Facial Expression				
	Discrimination (%)	Normal	Sad	Smile	Surprise	Angry
Edge detection method	Canny	100.0	91.3	99.4	95.7	94.8
	Laplace	93.2	87.1	89.5	85.6	87.4
	Sobel	63.4	49.7	66.8	64.9	62.6
	Robert	49.8	30.1	42.5	48.2	48.6

4. CLASSIFICATION OF IMAGES IN THE DATABASE WITH PARALLEL PROGRAMMING AND EXPERIMENTAL RESULTS

Use facial emotion classification was made using the software developed with visual C# parallel programming language on the distributed computer system architecture created within the framework of big data analytics of image files in the database. Sobel filter was used in the selection of facial expressions (Acharjya et al., 2012; Shrivakshan and Chandrasekar, 2012; Tsankashvili, 2018). Face selection was made thanks to the convolution matrices of the Sobel filter. The face image is divided into 4 equal parts and the edge detection processes are performed for eye and mouth detection. The results obtained at the end of the operations are assigned to the variables. Appropriate mood status was determined by comparing eye and mouth opening values with the values in the database. Before registering to the database, the analyzes were checked to ensure that the same results were not recorded in the database.

The database connection in the distributed computer system architecture was made using the Entity framework. This connection method is faster than adonet, so this is preferred for interrogation. A code blog has been developed to process the selected images in the processors and start the process. The process starts by selecting the image lines. Then, with the parallel for method, the images were taken to the threads and the processing lines were continued in this way. Face detection was done with the Sobel filter. Code section was created to take the eye areas.

A code section was prepared in which processes related to the mouth area were made. The code section was prepared, in which the database was checked and the records were made to the database. In the developed application, the working principle of facial emotion analysis was realized on these code blocks. Recording processes for emotion classification and data labeling were also carried out using these code blocks. The existing, previously known label values of the facial images in the database were compared with the labeling results obtained as a result of the classification calculations made in this study. The operation was carried out for 58 face image files registered in the database. The manual classification results of these were compared with the results obtained from the calculation in practice. The emotional categories obtained as a result of the classification of the facial images and the correct classification percentages piercing these categories are given in Table 2.

TABLE 2. EMOTION CATEGORIES OBTAINED AS A RESULT OF CLASSIFICATION OF FACE IMAGES AND CORRECT CLASSIFICATION PERCENTAGES CORRESPONDING TO THESE CATEGORIES.

	Facial Expression				
	Normal	Sad	Smile	Surprise	Angry
Correct classification percentages	95%	92%	90%	92%	86%

4.1. Application Developed for The Classification Of Image Files In The Database With Parallel Programming

The application is a Windows form application developed with Windows based visual C# parallel programming language. Visual Studio was used as IDE during the development phase. .NET 4.6 was used as the framework while developing in Visual Studio. There is no parallel programming library in frameworks before .Net framework 4. It is preferred because it contains libraries related to parallel programming in .Net framework 4.6 and it is one of the most current versions. Application file of the application's Windows desktop .exe extension has been created.

In the developed application, the select file button is used to start the processes. When the Select file button is pressed, it will open the My Documents-> Pictures file on the first boot as it is set as picture files in the OpenFileDialog toolbox. In later use of the

program, the most recently used file extension is opened. In the application, analysis of the desired number of images is performed by clicking the select file button. The window that opens and opens when the select file button is clicked is shown in Figure 2.

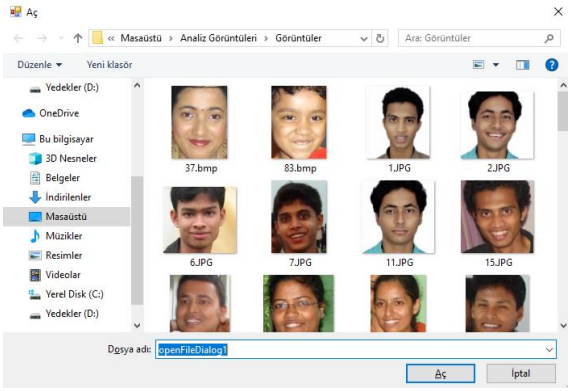

Figure 2. The window that appears when the Select File button is clicked.

The images selected from the opened window are processed in the processors of the computers in the distributed computer system architecture created randomly due to parallel programming; The name of the image is recorded with information such as mood, processing date. If one of the selected images was previously analyzed and saved in the database, another image is analyzed without analyzing the image again. The image processed during the analysis can be displayed on the form screen (PictureBox toolbar) for a short time.

The database is checked for the image currently processed in a computer processor. It is checked whether or not any action has been taken regarding the image. The first image related to the image is recorded as a result of the analysis. If the image is analyzed in another processor or on another computer in the distributed system while it is being processed, it is not saved again. The system prevents repetitive processes in the analysis process.

4.2. Classification Results of Images In The Database

By using parallel programming on the distributed computer system architecture, each image in the database is assigned to one of five different categories. The classification results of the images in the database are taken as in the figures shown in Figure 3.

When the operations are carried out in the architecture of the distributed computer system, data on their results are recorded in the database. Saving the result data about the transactions in the database provides the opportunity to be used for analysis. In the distributed computer architecture, information on which computer in the database image files are processed as threads is also recorded.

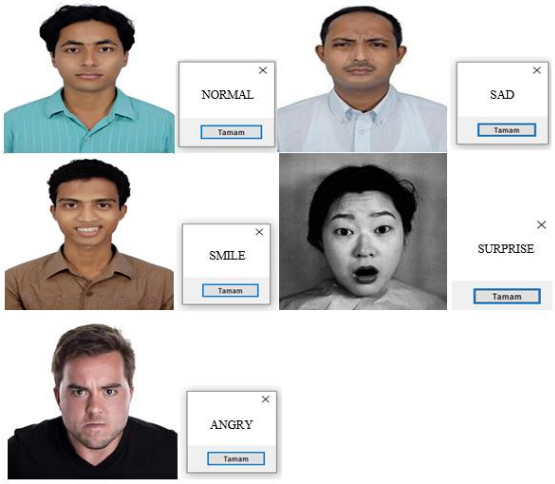

Figure 3. Classification results for each of the images in the image files in the database.

5. CONCLUSIONS AND DISCUSSIONS

In this research, distributed computer architecture, which acts as a high performance system, was created. Databases and parallel programming are used in distributed computer system architecture. Face image files in the database are classified for emotion analysis using the distributed computer system. It was concluded from the facial expressions that the formation of emotions was influenced by many factors. These factors can be divided into two environmentally and psychologically. Information was obtained about the psychological state of the person or the environment in which the person is directly affecting the emotions of the person. It is revealed that the person creates reactions with these gestures and gestures against these factors.

The importance of mimic movements was understood in the perception of emotions from facial expression. Based on these movements, we have been informed about what emotion the person has at that moment. The collected information about the reactions of the person according to his emotional state. A hundred datasets were created with this collected information. The data set contains images of different people with different emotional states. The emotions in the image were taken according to the test values of the application by taking the expressions in the mimics of the people. The results obtained are shown in the application success table.

Different programming languages and methods were used in this study. The programming languages used are Matlab and visual C# parallel programming language. In the software prepared for the study subject: Matlab program was used to process image files and simplify the images. Visual C# predicts facial emotional states from parallel programming image files and has been applied in the classification process.

Visual C# parallel programming coding and classification methods and performances of algorithms are used. Threads created using the basic distributed computer system architecture were processed on other computer processors on the common computer network. Thanks to the filters used in basic distributed computer architecture, the distortion and noise in the data are minimized and classification results are obtained with higher accuracy percentages. In this study, sobel filter was used. With the sobel filter used, realistic emotion values were estimated in the facial emotion classification process. High accuracy rates were obtained in the classification results. In the next study, performance comparisons of Sobel filter and other filters in image processing will be made. In addition, the performance of the deep learning method will be investigated in facial emotion analysis.

Acknowledgments

We would like to thank the editor and three anonymous referees for their valuable comments and suggestions on the manuscript.

REFERENCES

Acharjya, Pinaki Pratim; Ritaban Das and Dibyendu Ghoshal (2012). Study and Comparison of Different Edge Detectors for Image Segmentation. Global Journal of Computer Science and Technology Graphics & Vision Volume 12 Issue 13 Version 1.0 Online ISSN: 0975-4172 & Print ISSN: 0975-4350.

Cliff, Ashley; Jonathon Romero, David Kainer, Angelica Walker, Anna Furches and Daniel Jacobson (2019). Article A High-Performance Computing Implementation of Iterative Random Forest for the Creation of Predictive Expression Networks. Genes, 10, 996; doi:10.3390/genes10120996.

Coulouris, George; Jean Dollimore, Tim Kindberg and Gordon Blair (2012). Distributed Systems: Concepts and Design Fifth Edition. Addison-Wesley. 1067 pages.

Czarnul, Paweł; Jerzy Proficz and Krzysztof Drypczewski (2020). Survey of Methodologies, Approaches, and Challenges in Parallel Programming Using High-Performance Computing Systems. Scientific Programming. Volume 2020, Article ID 4176794, 19 pageshttps://doi.org/10.1155/2020/4176794.

Dordal, Peter L. (2020). An Introduction to Computer Networks Release 1.9.21. http://intronetworks.cs.luc.edu/current/ComputerNetworks.pdf, 896 pages.

Ezhilraman, S. Vahini and Sujatha Srinivasan (2018). State of the art in image processing & big data analytics: issues and challenges. International Journal of Engineering & Technology, Volume 7, Pages 195-199.

Hager , George and Gerhard Wellein (2011). Introduction to High Performance Computing for Scientists and Engineers. Chapman & Hall/CRC Computational Science Series. 349 pages.

Hung, Jason C., Neil Y. Yen and Kuan-Ching Li (2016). Frontier Computing: Theory, Technologies and Applications. Springer (Lecture Notes in Electrical Engineering), 1252 pages.

Kouanou, Aurelle Tchagna; Daniel Tchiotsop, Romanic Kengne, Djoufack Tansaa Zephirin, Ngo Mouelas Adele Armele and René Tchinda (2018). An optimal big data workflow for biomedical image analysis. Informatics in Medicine Unlocked. Volume 11, Pages 68-74.

Lerman, Julia (2009). Programming Entity Framework. O'Reilly Media, Inc. 731 pages.

Murdocca, Miles J. and Vincent P. Heuring (1999). Prıncıples Of Computer Archıtecture. Prentice Hall. 654 pages.

Shrivakshan, G. T. and C. Chandrasekar (2012). A Comparison of various Edge Detection Techniques used in Image Processing. IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 1, September 2012 ISSN (Online): 1694-0814 www.IJCSI.org.

Tsankashvili, Nika (2018). Comparing Edge Detection Methods. https://medium.com/@nikatsanka/comparing-edge-detection-methods-638a2919476e.

Yadav, Shikha; Preeti Dhanda and Nisha Yadav (2013). High Performance Computing through Parallel and Distributed Processing. Journal Of Harmonized Research in Engineering 1(2), 54-64.