International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-6, March 2020

Load Balancing at Fog Nodes using Scheduling Algorithms

Mujthaba GM, Manjur Kolhar, Abdalla AlAmeen

Abstract*: Cloud Computing proves to be most predominant innovative field in the area of Information technology. Cloud is best suited for small scale to large scale businesses and personal purposes such as storing, computing, managing data & resources, running applications and many more. Due to increasing large volumes of data over cloud servers created subsequent specific issues like data maintainability, network elasticity, managing Internet of Things (I.o.T's) devices and many more. Recent progresses in Technology are given rise to fog computing or decentralized cloud to overcome cloud server issues called fog nodes. In this paper we present a brief note on how cloud issues can overcome using fog nodes benefits along with elaboration of load balancing factor. To maintain load balancing of fog nodes no much appreciable work took place in the field of fog computing. This paper proposes a scheduler which receives the devices in to a Job Queue to be connected over cloud. To apply scheduling algorithms like F.C.F.S, S.J.F, P.S, R.R and W.R.R. over fog nodes will be discussed along with their merits & demerits. At last we try to compare the various parameters of load balancing among various scheduling algorithms. In this paper we focus on how fog nodes perform functions like considerable storages, low latency, heterogeneity, allocation & interaction with limited IoT devices and Security along with architecture cloud to fog. During allocation of IoT devices to various fog nodes we will come across a serious issues i.e load balancing on fog nodes. Our detailed study presents the comparison of above mentioned scheduling algorithms load balancing factors such as rich resources allocations & Balancing among fog nodes, Identification of devices, Authentication of fog nodes, bandwidth consumption, location awareness, response time, cost maintenances, Intrusion detection, fault forbearances and maintainability.

Keywords: First Come First Serve (F.C.F.S), Shortest Job First (S.J.F), Priority Scheduling (P.S.), Round Robin (R.R.), Weighted Round Robin (W.R.R), Internet of Things (I.o.T's),

Manuscript received on March 15, 2020.
Revised Manuscript received on March 24, 2020.
Manuscript published on March 30, 2020.

* Correspondence Author

Mr.Mujthaba Gulam Muqeeth* Computer Science Department, College of Arts and Science, Prince Sattam Bin Abdul-Aziz University,Wadi Al Dawassir, Riyadh, K.S.A. Email: mujthaba.gmm@gmail.com

Dr.Manjur Kolhar Computer Science Department, College of Arts and Science, Prince Sattam Bin Abdul-Aziz University, Wadi Al Dawassir, Riyadh, K.S.A. Email: manjur.kolhar@gmail.com

Dr.Aballa AlAmeen, Computer Science Department, College of Arts and Science, Prince Sattam Bin Abdul-Aziz University, Wadi Al Dawassir, Riyadh, K.S.A. Email: abda711su@gmail.com

I. INTRODUCTION

Cloud is said to be most grown reliable medium for transmission of data over a network to a million of users for providing appropriate striking services to business and personal use or end users [1][2]. Numerous vendors exits in the market to provide cloud services like Amazon EC2, Microsoft Azure, Google cloud and many more to meet small scale to large scale organization demands[3].Cloud is a platform where it interconnects Nemours data servers for providing benefits like elasticity pay as per use ,rate dropping strategies, reducing complexities, un limited storage & access, out sourcing data, robust security , flexibility, scalability, virtual machines, usage of better hardware & services, accessibility of future expected infrastructure & services and so on[3]. Each data server on cloud is allocated with countless Internet of Things (IoT's) or devices which increases the load over cloud. This exponential growth of load on cloud arises to storms

A. Cloud Issues

As stated in [4] although Cloud computing has now occurred to become one of the finest technology for personal, organizations and business applications. Millions of users & devices are connected to cloud by means of Infrastructure, Platform and services. Due to increasing load on the cloud certain issues and problems associated with cloud.

- Outsourcing data among various cloud servers
- Managing data & resources, network bandwidth
- Configuring computing resources
- Accessing distant servers,
- Provision of hardware infrastructures
- Managing millions of geographically distributed nodes
- Telecommunication links
- Upkeep costs
- Fault tolerance
- Security
- Power full cryptographic algorithms
- Data tampering.

B. Why Fog

As discussed in [3] to covenant these issues technology given rise or extension to Fog Computing or fogging or decentralized clouds by CISCO routers. Fog is emerging trend in the field of communication which implements cloud services at the edge of a cloud termed as fog nodes. Each fog node is allocated considerable users to implement their services like as follows

- Storing & accessing data
- Use network
- Control functions

- Communication between various data servers
- Managing resources
- Computational capabilities

C. Fog Benefits

The only key difference which varies fog from cloud is fog computes services at the edge of network. The significance of using fog nodes due to its inimitable gains likes as follows as come up in [3]

- Substantial storages
- Low latency
- Heterogeneity
- Allocation & interaction with limited IoT devices
- Monitoring IoT's & proximity to end users
- Effortlessness
- Abstraction
- Network elasticity
- Supporting virtual networks
- Tough security
- Fault tolerance
- Limitations on outsourcing data
- Managing infrastructure or hardware devices.

D. Need of Load Balancing at Fog nodes

Due to exponential growth of end users on fog nodes give raises plentiful serious concerns or issues affecting to load balancing on fog nodes which are listed below as discussed in[1][6]and [7]:

- Balancing the load of various IoT's on fog nodes
- Allocation of resources to various IoT's or devices by fog nodes
- Maintaining network elasticity due to mobility of devices from one fog node to another
- Identification or tracing various IoT's
- Estimation of considerable bandwidth due to device mobility
- Location awareness
- Intrusion Detection
- Maintaining data integrity, conviction & privacy
- Authentication of various IoT's by fog nodes

To address above mentioned specific issues no much considerable research took place except few papers published to maintain load balancing at fog nodes.

E. Need of scheduling algorithms over cloud or fog nodes

As the cloud consists of mixed environment of hardware & network resources, platforms for various applications or software resource and many IoT's or devices. Each fog node is connected with various devices which request or releases the resources. To maintain the requests of resources by various devices need to maintained and process in proper mechanism. Each scheduling algorithm is consists of various advantages and disadvantages. This paper incorporates the scheduling algorithms which used to maintain or balance the load at cloud data servers. The detailed description of scheduling and SWARM algorithms is explained in the methodology. The main objective of the algorithms to benefit the fog nodes as mentioned below [8]

- Improves the efficiency of fog nodes
- Allocate resources among fog nodes
- Maximize utilization of resources
- Calculation of response times and waiting time among fog nodes
- Maintaining network elasticity
- Priority among fog nodes
- Predicting fault tolerance and managing Infrastructure
- Inheriting Computational Intelligence based on ant colonies

II. LITERATURE REVIEW

Cloud is proved to be most suited technology to interconnect various networks and their devices [1]. It interconnects the millions of devices in order to build the applications as pay per use. Users are provided with Infrastructure, Platform and service to run & deploy their applications [1][2]. Several vendors exist in the market who provide cloud services to their clients like Amazon EC2, Microsoft Azure, and Google Cloud to meet their demands. Cloud is extensively rich in providing services like unlimited data storage, running applications, virtual systems, security, robust ness and etc[3]. Exponential growth of users and their data leads the cloud to pass certain limitations. Such issues like data management over data centers, maintain network bandwidth, effective use of resources, configuring routers, directing servers, maintain security, Integrity, confidentiality, fault tolerance, effective cryptographic algorithms and many more.

To eradicate these cloud issues like balancing the devices, technology given rise to fog computing. Cloud distributes its devices or IoT on various fog nodes. Each fog node is responsible for allocating and revoking the services to the cloud. Managing the devices on each fog nodes properly is not easy means of load balancing. To handle the load balancing this paper focus various scheduling algorithms like FCFS, SJF, PS, RR and WRR and so on [9][10][11].These algorithms mainly deals with balancing the fog node properly in terms of resources allocations & Balancing among fog nodes, Identification of devices, Authentication of fog nodes, bandwidth consumption, location awareness, response time, cost maintenances, Intrusion detection, fault forbearances and maintainability and so on.

III. METHEDOLGOY TO IMPLEMENT SCHEDULING ALGORITHM

As stated in [4][12] and [13] usually cloud schedulers are used to manage the resources (Infrastructure, Platform and services) over cloud. Due to overload of devices, Cloud decentralized into fog nodes. Each fog node is assigned with limited devices to accomplish same functionalities like cloud on limited devices. Allocation of devices to the respective fog nodes is became a serious concern or issue. To overwhelm these issue this paper presents a scheduler which regulates the devices based on scheduling algorithms. Scheduler main aim is to allocate the devices to respective fog nodes based on the number of devices connected to it. The below architecture explain how scheduler is used to schedule the devices to fog nodes when arrived in job queue.

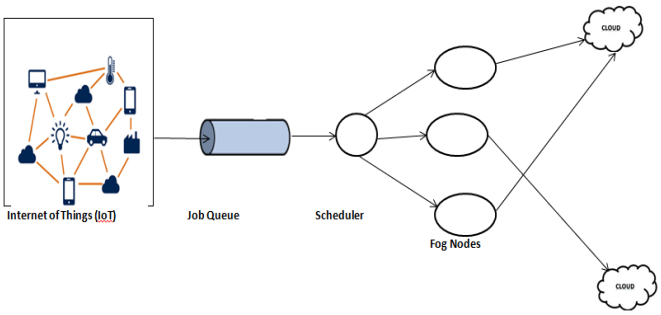

Fig 1 Scheduling of Devices to Fog nodes over Cloud

As said in the above fig 1 the main role of scheduler to assign to assign the fog nodes depending upon their waiting time, length, burst time and priority. Be contingent to these parameters its is upto the scheduler to assign implement the which scheduling algorithm is best suited to assign which node .

A. Notations used for scheduling devices

Cs: Set of Cloud Servers Cs={Cs1 , Cs2, Cs3…….. Csn} such that $\sum_{i=1}^{n} Csi$

Fi: Set of fog nodes F= { F1, F2, F3 ---- Fn} such that $\sum_{i=1}^{n} Fi$

Di: Set of IoT's or Devices D ={ D1,D2, D3 ……. Dn} such that $\sum_{i=1}^{n} Di$

B_T (D_i): the duration for which a device gets control over the Fog node (Fi) defined as device burst time

A_T (D_i): The Arrival time of devices Di at their respective fog nodes Fi

JQ: Place the devices in Job Queue and Initialize its flag bit to "False" before it is connected to any fog node else set to true.

RQ: If any Di in the JQ is detached or removed or blocked due to any reason then that Di will added to Ready Queue (RQ).

TQ(Di): Time Quantum is to ensure no Di can obtain more than one time slice assign with any fog node Fi

pi: Priority or Medium priority (mpi) of Di such that Di (pi)

hpi : Highest Priority for Di (hpi) and lpi : Lowest Priority for Di (lpi)

TAT(Di): Turnaround time for any $\sum_{i=1}^{n} Di$ i is the amount of time consumed to full fill request.

CT(Di): Completion time for any $\sum_{i=1}^{n} CT(Di)$

AWT(Di) :Average Waiting time for any $\sum_{i=1}^{n} AWT(Di)$

WT(Di) : Waiting time for any Di such that

$\sum_{i=1}^{n} WT(Di) =$ CT(Di)- (AT (Di)+ BT (Di))

Clear explanation and demonstration of how these scheduling algorithms works over schedulers of fog nodes are demonstrated below:

IV. SCHDEULING ALGORITHMS

Scheduling is a process which allows each device to connect with respective fog nodes. Various scheduling algorithms used over cloud to assign the devices to each fog nodes. Below are few scheduling algorithms discussed?

A. Applying F.C.F.S to cloud environment

First come, first served (FCFS) as discussed in [11] is a scheduling algorithm which is applied to route the networks or route management. The primary idea behind this algorithm is to place the process with in the job queue as per their arrival. Processor intern execute the process according to their sequence of arrival at queue (First come First Serve).Same concept of FCFS route scheduling algorithm will be applied to fog nodes. Detailed description of FCFS algorithm deployed over fog nodes can be given as follows.

Step 1: Each of the fog nodes will be assigned with IoT's according to their arrival in the job queue JQ.

Step 2: Given n devices Di with their burst times BT (Di) and arrival time AT(Di)

Step 3: The task is to find average waiting time AWT(Di) and average turnaround time TAT(Di) using FCFS scheduling algorithm.

Step 4: Stop if the entire Di is finished in JQ and assigned to fog nodes.

Table- I: F.C.F.S for cloud nodes

Devices or IoT's needs to connect to fog nodes	Burst time/ Time to assign to nodes	Arrival time of Di to JQ
D1	20	1
D2	17	2
D3	3	3

Waiting time for the devices is $\sum_{i=1}^{n} Di$ WT (D1) = 0 WT (D2) = 20 and WT (D3)= 37

Average waiting for device is $\sum_{i=1}^{n} Di$ is (WT (D1)+ WT (D2)+ WT (D3))/ 3 = (0 +20+37)/3 = 19

Inference: Main defect of this FCFS algorithm is that the any Di needs to wait for a longer time even though it has small burst time. Each device get a chance to get connect with fog nodes so there will be no starvation.

B. S.J.F. over cloud environment

Assigning the devices with increasing (Initiate with smallest burst time or low priority) to fog nodes. It causes good advantage to devices having minimum burst time to connect the fog nodes. The detailed description of the devices entering the job queue with shortest bust time will be connected first to corresponding fog nodes as shown on the below table:

Step 1: Place all the devices Di in a JQ

Step 2: Choose the device Di with SJF (minimum burst time).

Step 3: Assign the Di to fog node depends on his processing capabilities

Step 4: Connect these Di to their respective fog nodes until JQ is empty

Step 5: calculate waiting time and average waiting time.

Table- II: S.J.F. for cloud nodes

Devices or IoT's needs to connect to fog nodes	Burst time/ Time to assign to nodes	Arrival time of Di to JQ
D1	20	1
D2	17	2
D3	3	3

Waiting time for the devices is $\sum_{i=1}^{n} Di$ WT(D1) = 0 WT(D2) = 17 and WT(D3)= 20

Average waiting for device is $\sum_{i=1}^{n} Di$ is (WT(D1)+WT(D2)+WT(D3))/ 3 = (0 +17+20)/3 = 37/3 = 12.33

Inference: As shown in the above table if FCFS algorithm is implemented over the fog nodes on cloud it could be very much beneficial for the I.o.T's who come fist in to the queue .These devices may get granted with the resources they request over the cloud. Generally it is not suitable for the IoT's with shortest waiting time because they can't be processed before the ioT's connected to them. Hence the Average waiting time and waiting time for devices or device with shortest priority will have an over advantage. Throughput increases as more devices can connect within shortest time.

C.P.S Over fog nodes or cloud environment

To overwhelm the defect of FCFS each device Di is assigned with some priority depending upon its B.T. Devices with highest priority BT will be assigned first to the fog nodes and later on decreasing priorities. Using this algorithm will eradicate the issue of waiting time

Inference: Best suited for the devices with highest priority. Implementation of priority based scheduling algorithm is fair an easy but assigning the priorities to the devices Is another serious concern.

D. R.R over cloud environment

This algorithm is discussed in [9] considered as most prudential network scheduler in the field of computing technology. The key role of RR is to balance the load requests between numbers of processes being employed. Each process will be assigned or allocated with a Quantum of time or time slice and get executed when it turn knocks. Same idea will be applied on cloud architecture or network which consists of several cloud servers. Each server is assigned with number of fog nodes which further allocated with various IoT's or devices. Scheduling the devices to the fog nodes will be considered as another serious issue which can be resolved by using RR algorithm. This paper will consider how RR algorithm can be beneficial in allocating the devices to respective fog nodes as given below

Algorithm RR_ Fog Nodes

Step 1: Let Input the number of Cloud servers { Cs1, Cs2, Cs3 …… Csn} ,fog nodes { F1, F2, F3 ---- Fn} and IoT's or devices { D1,D2, D3 ……. Dn} Connected in such a way that Di ∈ Fi ∈ Csi where i<=n

Step 2: Input BT (Di) and AT (Di) for each device connected to their respective fog node such that Di ∈ Fi

Step 3: Initialize and set the flag bit to "false" for each device Di before it is allocated to any Fi . Place Di JQ

Step 4 : Input TQ for each Di

Step 5: Each Di will be connected to Fi based on priority. Priority can be defined by TQ Such that hpi =(TQ+20% * TQ))lpi = (TQ –(20% * 10)) and(mpi) =pi

Step 6: If there is any Di with very low BT (Di) in the RQ then assign that BT (Di) to the fog Fi and finish its allocation. Set its flag bit to TRUE in the JQ and calculate it's WT (Di) and TAT (Di) and remove Di where i<=n from the RQ.

Step 7: Else assign the Fog with other device Di such that 1<=n.

Step 8: If (B_T (Di) <= $T_{Q(}$Di)) then allocate the entire Di for its remaining B_T (Di). Set its flag to TRUE and Calculate its W_T(Di) and turnaround time and remove it from the RQ (the TQ is different for different priority processes).

Inference: This algorithm is best suited for improving average response time. Each device is limited particular amount of time to hold in the job queue after that will be moved at the back of queue. Due to round robin (Each device with equal quantum of time) there is a minimal chance of occurring starvation.

E. W.R.R over cloud environment

The basis for WRR algorithm as said [9] and [10] is RR and various scheduling algorithms like priority scheduling algorithms. The key idea behind WRR is to retain the merits of RR by eliminating its errors and integrate various priority scheduling algorithms. If we consider RR the main error is that each process to be executed by the processor given equal amount of time and priority in the queue. Being same time slice and priority to all the process cause them to retain in the queue without execution. Same concept of WRR will applied over the cloud which is interconnection of various fog nodes. To balance the load balance issue of various IoT's among the fog nodes will be considered as serious concern. To overwhelm this issue each device will be assigned with priority or an integer value or weight before being connected to fog nodes.

Algorithm WRR_FogNodes

All new jobs are entered into the job queue

Step 1: Let devices place in a JQ before connect to a fog nodes in such a way that Di ∈ Fi ∈ Csi where i<=n

Aging: All jobs are evaluated if an increase in weight is needed

Step 2: Assign each device Di with some priority pi or weight which defines aging of each device Di

Step 3 Place the Di with highest priority at the front of queue such that devices with highest priority can be assign first their respective fog nodes such that Di (hpi) > ….>Di(mpi) >…..> Di (lpi)

Step 4: When highest priority device Di is assigned to fog nodes front moves to other Di based on the next priority.

Step5: Based on the priority all the devices are connected to fog nodes till the JQ is empty.

Job queue reordered with higher weight at front

Inference: WRR incorporates the features both round robin and priority based scheduling algorithm. It mainly abolishes the problem of starvation. This algorithm needs to be verified by fairness, responsiveness, and efficiency.

V. RESULTS AND OBSERVATIONS

As sated in the below table following observations or finding s made upon the implementations of scheduling algorithms Table III Observations of Scheduling algorithms over fog nodes

Name of the scheduling algorithm	Advantages	Limitations
F.C.F.S	No starvation. Each device gets a prospect to get connect with the fog nodes.	The devices with the low burst time need to wait for a long time.
S.J.F	Best suited for the devices with the low burst time. Each device enters the queue can execute or processed quickly if its burst time is short.	Not suited for the devices which need to be processed for a long time by the fog nodes
P.S	It is best suited for the devices with highest priority.	The devices with normal or low priority need to wait for a long time even though they enter earlier with in the queue.
R.R	This algorithm is well suited for to balance the load among the fog nodes irrespective to their burst time. Each device will be processed with same amount of time.	It is not best suited for the devices with the low burst time as it makes the fog nodes to be idle for more time.
W.R.R.	It incorporates the advantages of both priority and round robin algorithm.	Strong mechanism needed to verify fairness, responsiveness and efficiency of the devices to be connected to the fog nodes.

VI. CONCLUSIONS

In this paper we conferred how scheduling algorithm can be used on cloud environment to balance the load at various fog nodes. Each scheduling algorithm can be deployed over fog nodes to maintain the load balancing over cloud computing environment. In this paper we focuses on benefits ,challenges and limitations of cloud issues Appropriate selection of scheduling algorithm to deploy over fog nodes is important and mandatory issue. Prevailing scheduling algorithms presents fair throughput and also cost operative but will not be much effective relating to accessibility and consistency. Due to this issues there is an immense need to revise the scheduling algorithms to deploy over fog nodes. Main objective of revising the scheduling algorithm is to have proper selection of algorithm to grant and release the resources to the devices connected over fog nodes. Balance the load over fog nodes by switching the devices to other idle fog nodes is another important concern. To deploy which algorithm over fog nodes to load balance is the main objective of the scheduler. To perform this task the scheduler much keep in concern about the responsiveness, fair working principle, throughput, burst time, average waiting time before allocating the devices to the respective fog nodes. Still there is much scope for the researchers to implement these algorithms over schedulers at fog nodes and obtain the fair complexity of each algorithm.

REFERENCES

1. Dave, A., Patel, B., & Bhatt, G. (2016, October). Load balancing in cloud computing using optimization techniques: A study. In Communication and Electronics Systems (ICCES), International Conference on (pp. 1-6). IEEE.
2. Nath, D. (2017). A Survey on various Load Balancing Algorithms in Cloud Computing. International Journal of Advanced Research in Computer Science, 8(5).
3. Moreno-Vozmediano, R., Montero, R. S., Huedo, E., & Llorente, I. M. (2017). Cross-Site Virtual Network in Cloud and Fog Computing. IEEE Cloud Computing, 4(2), 46-53.
4. Sookhak, M., Yu, F. R., Khan, M. K., Xiang, Y., & Buyya, R. (2017). Attribute-based data access control in mobile cloud computing: Taxonomy and open issues. Future Generation Computer Systems, 72, 273-287.
5. Chiang, M., Ha, S., Chih-Lin, I., Risso, F., & Zhang, T. (2017). Clarifying Fog Computing and Networking: 10 Questions and Answers. IEEE Communications Magazine, 55(4), 18-20.
6. Puliafito, C., Mingozzi, E., & Anastasi, G. (2017, May). Fog Computing for the Internet of Mobile Things: issues and challenges. In Smart Computing (SMARTCOMP), 2017 IEEE International Conference on (pp. 1-6). IEEE.
7. Alrawais, A., Alhothaily, A., Hu, C., & Cheng, X. (2017). Fog Computing for the Internet of Things: Security and Privacy Issues. IEEE Internet Computing, 21(2), 34-42.ha
8. Thakur, P., & Mahajan, M. (2017). Different Scheduling Algorithm in Cloud Computing: A Survey. International Journal of modern computer science, 5(1).
9. Parekh, H. B., & Chaudhari, S. (2016, December). Improved Round Robin CPU scheduling algorithm: Round Robin, Shortest Job First and priority algorithm coupled to increase throughput and decrease waiting time and turnaround time. In Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC), 2016 International Conference on (pp. 184-187). IEEE.
10. Fang, Yiqiu, Fei Wang, and Junwei Ge. "A task scheduling algorithm based on load balancing in cloud computing." International Conference on Web Information Systems and Mining. Springer, Berlin, Heidelberg, 2010.
11. Raghu, G., Sharma, N. K., Domanal, S. G., & Reddy, G. R. M. (2018). Memory-Based Load Balancing Algorithm in Structured Peer-to-Peer System. In Progress in Intelligent Computing Techniques: Theory, Practice, and Applications (pp. 431-439). Springer, Singapore.
12. Havet, Aurelien, et al. "GenPack: A generational scheduler for cloud data centers." Cloud Engineering (IC2E), 2017 IEEE International Conference on. IEEE, 2017.
13. Guevara, Judy C., Ricardo da S. Torres, and Nelson LS da Fonseca. "On the classification of fog computing applications: A machine learning perspective." Journal of Network and Computer Applications (2020): 102596.

AUTHORS PROFILE

Mr. Mujthaba Gulam Muqeeth holding B.Tech (Kakatiya University), M.Tech (J.N.T.U.H) with specialization in Computer Science and Pursuing is PhD in Artificial Intelligence. Having a vast experience of 17years in the teaching field of computer science engineering courses in various reputed engineering colleges at both national and international level. Presently working as a Lecturer of Computer Science Department at Prince Sattam Bin Abdul-Aziz University, kingdom of Saudi Arabia since nine years. Besides teaching working with Quality Control Department accredited with N.C.A.A.A. His area of interest includes Cloud Computing, Software Engineering and Artificial Intelligence. He published papers and books on Cloud Computing and Computer Networks.

Dr. Manjur Kolhar did his Master of Application master degree and Ph.D from Karnataka university, Dharwad, Karnataka, and Malaysia respectively. Presently he is working as an Associate Professor of Computer Science Department at Prince Sattam Bin Abdul-Aziz University, Kingdom of Saudi Arabia. He has more than 20 years of experience in software development and academics. His area of interest includes Computer Networks Cloud Computing, Software Engineering, Artificial Intelligence and Neural Networks. He published various papers and books on his research areas. Presently on working image processing, focused on smart parking, using convolution network. He has received various grants from deanship of research, Prince Sattam Bin Abdulaziz University.

Dr. Abdalla Al Ameen holding Ph.D in Computer Science is working as an Associate Professor and Head of Computer Science Department at Prince Sattam Bin Abdul-Aziz University, K.S.A. His area of interest includes Databases Computer Networks Cloud Computing, Software Engineering and computer security. Author had vast experience working in gulf international universities. He much acquainted with quality control unit of NCAAA and other accreditation bodies. He published various papers and books on his research areas in prestigious ISI journals .