iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 **Issue:** VI **Month of publication:** June 2018

DOI: http://doi.org/10.22214/ijraset.2018.6182

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue VI, June 2018- Available at www.ijraset.com

Distribution of Task using Smartphones for Server Load Balancing

Supriya Rajhans[1], Bhagyashree Shitole[2], Shamal Raut[3],Shivkanya Barhate[4] ,P.S Shinde(Assistant Professor)[5]
[1,2,3,4] *Department of Engineering, Zeal College of Engineering and Research,Pune*

Abstract: ***Android devices have become a popular and —anywhere, everywhere‖ computational resource for a wide range of requirements. Due to its _mobile‘ nature, it allows people to carry high computational power in their hands where the computational power is comparable against that of a desktop or laptop. This computational power i.e CPU, Storage Memory and RAM of them is almost same like a desktop computer or Laptop in the recent years. However, Android devices were not being used for executing any computation intensive tasks till 2015 extensively. A recent study shows that users keep their Android devices idle for 8 hours on an average in the night time at their home. During this period, the device would be idle for most of the times except that it needs some CPU cycles and memory for download and/or other user requested applications running in the background. This processing power can be utilize for task scheduling. Android devices follows greedy algorithm for scheduling the tasks and faces a unique set of technical challenges due to the heterogeneity in CPU clock speed, variability in network bandwidth, and lower availability than servers. This paper uses a new scheduling algorithm (Linear Programming Algorithm) and we addressed many of these challenges to develop a distributed computing infrastructure using smart phones for task of Image Processing.***

Keywords: Distributed System; Mobile Computing; Boot- strapping; Linear Programming; Load Balancing; Image Processing;

I. INTRODUCTION

The main aim is to develop a task distribution system on cloud using smartphones for image processing. In this system we are using linear programming to distribute the image for processing. At night or when smartphones are not in use, at that time their processing power and storage will become wastage.

These smartphones can be used as small volunteers and the required task is distributed on these volunteers so that they compute different parts separately and at the end collectively gather the output. The problem can be justified as to use idle state smartphones for task distribution and computing on the basis of processing speed of smartphones , which will give cost effective and load balancing alternative to desktop system ,also reduces the load on central server.

The main goal of the product is to create a distributed system where there is use of idle state mobile phones for completion of particular task. This system will al- low the user a platform where they can select a task for computation and distribute it over the volunteers i.e smartphones.

It will also enable user to save the cost of computation. since there is use of linear programming for task distribution used so, it will be give us fast processing through multiple mobile phones working parallel y. While distribution of tasks it also considers storage capacity and computational capacity of each volunteer. For distributed computing now .we have used Image processing as a task ,but in future this can be used for any data and task. Also there may use of both computers and smartphones together.

A. Detail System Requirement Specification

This document specifies the software requirements of Task Distribution on Cloud Using Smartphones for Im- age Processing .This SRS specifies the entire system be- ing developed and tries its best to describe everything. Lots of mobiles have idle state at night and while not in use.

Therefore there is wastage of processing and efficient features smartphones. The Processing power of smartphones can be utilize by distribution of tasks. Also we need improved speed of processing.

Thus this system provides us cost effective distribution of tasks over smartphones. There is re-usability of idle state processors.

This SRS covers the specification of different modules that are contained in the application. Modules are namely Task Submitter, Task Processing and Server .

B. Product Perspective

Android devices (Phone, Tablet) are steadily gaining popularity and computational power (CPU, Storage Mem- ory and RAM) of them is almost on par with a desktop computer or Laptop in the recent years. However, An- droid devices were not being used for executing any computation intensive tasks till 2012 extensively. A recent study shows that users charge their Android de- vices for 8 hours on an average in the night time at their home. During this charging period, the device would be idle for most of the times except that it needs some CPU cycles and memory for downloads and/or other user requested applications running in the back- ground. Existing system which leverages the hardware resources of the idle / unused Android devices follows greedy algorithm for scheduling the tasks.

Every enterprise in the world has the basic need of fast server to work efficiently as well as rapidly. But day b day computational workloads on server in the form of multiple task requests are also increasing. Because of these multiple tasks on cloud server, it works slower and sometimes gets failed. In this present situation dis- tributing or scheduling of tasks on the distributed computing system is the only solution to reduce workload on server. To form the best distributed computing sys- tem we are using smart phones as volunteer or android clients.

Smart phones are energy-efficient and cost-effective alternative to running certain tasks of traditional servers. When these smart phones are plugged into power source for charging battery then this idle phones provides the increasing computing capabilities and sizable computing infrastructure. Distributing tasks for computations must utilize all the resources equally, no resource should be under or over utilize, and this problem leads to focus on the load balancing technique to support the cloud for processing tasks. In this paper we are proposing the load balancing frame work as cloud supporter for processing tasks on smart phones when they are plugged into the power sources. Lots of mobiles have idle state at night and while not in use. Therefore there is wastage of processing and efficient features of smart phones. The Processing power of smart phones can be utilize by distribution of tasks. Also we need improved speed of processing. Thus this system provides us cost effective distribution of tasks over smart phones. There is re-usability of idle state processors. This system will allow the user a platform where they can select a task for computation and distribute it over the volunteers i.e. smart phones. Since, linear programming is used for task distribution and multiple mobile phones are working in parallel, the processing become faster. For distributed computing, we have used Image processing as a task.

II. PROPOSED WORK

Android devices (Phone, Tablet) are steadily gaining popularity and computational power (CPU, Storage Memory and RAM) of them is almost on par with a desktop computer or Laptop in the recent years. However, Android devices were not being used for executing any computation intensive tasks till 2012 extensively. A recent study shows that users charge their Android devices for 8 hours on an average in the night time at their home. During this charging period, the device would be idle for most of the times except that it needs some CPU cycles and memory for downloads and/or other user requested applications running in the background. Existing system which leverages the hardware resources of the idle / unused Android devices follows greedy algorithm for scheduling the tasks. Every enterprise in the world has the basic need of fast server to work efficiently as well as rapidly. But day by day computational workloads on server in the form of multiple task requests are also increasing. Because of these multiple tasks on cloud server, it works slower and sometimes gets failed. In this present situation distributing or scheduling of tasks on the distributed computing system is the only solution to reduce workload on server. To form the best distributed computing system we are using smart phones as volunteer or android clients. Smart phones are energy-efficient and cost-effective alternative to running certain tasks of traditional servers.

When these smart phones are plugged into power source for charging battery then this idle phones provides the increasing computing capabilities and sizable computing infrastructure. Distributing tasks for computations must utilize all the resources equally, no resource should be under or over utilize, and this problem leads to focus on the load balancing technique to support the cloud for processing tasks. In this paper we are proposing the load balancing frame work as cloud supporter for processing tasks on smart phones when they are plugged into the power sources. Lots of mobiles have idle state at night and while not in use. Therefore there is wastage of processing and efficient features of smart phones. The Processing power of smart phones can be utilize by distribution of tasks. Also we need improved speed of processing. Thus this system provides us cost effective distribution of tasks over smart phones. There is re-usability of idle state processors. This system will allow the user a platform where they can select a task for computation and distribute it over the volunteers i.e. smart phones. Since, linear programming is used for task distribution and multiple mobile phones are working in parallel, the processing become faster. For distributed computing, we have used Image processing as a task.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue VI, June 2018- Available at www.ijraset.com

III. SYSTEM DESIGN AND ARCHITECTURE

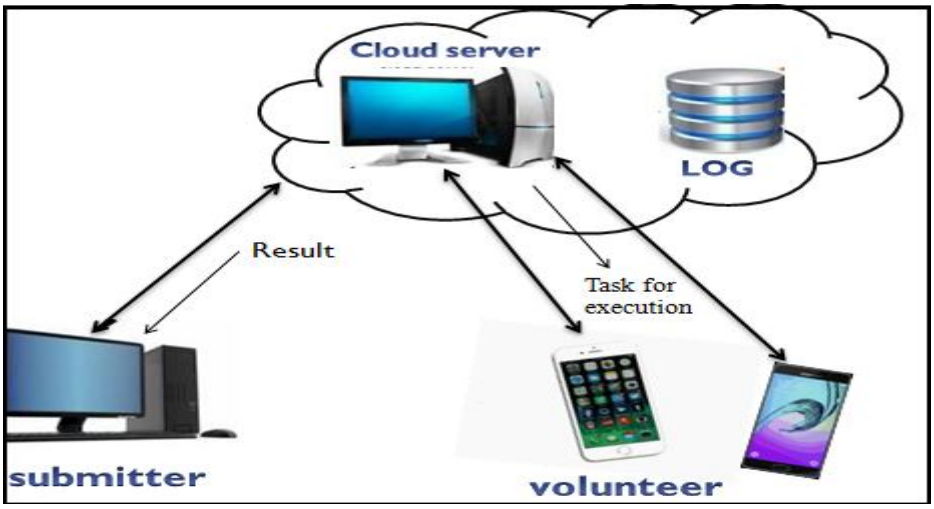


Fig.1: Architecture Diagram

The proposed system will have three components like Job Submitter, Cloud Server and Volunteers. Fig 1 shows block diagram for system.

A. Job Submitter

1) The Job Submitter will submit the task to server.

2) The task can be any time consuming which actually takes time to execute such as file text/doc, image processing etc.

B. Cloud Server

1) After receiving of the task, server will distribute the task to volunteers.

2) The load balancing algorithm like linear programming can be used at the server side. Based on current load the task will be submitted to volunteers.

C. Volunteers

1) Volunteers will complete the task processing and then send the response to server.

2) Server will integrate the result and send the reply back to Submitter application.

III.FLOWCHART

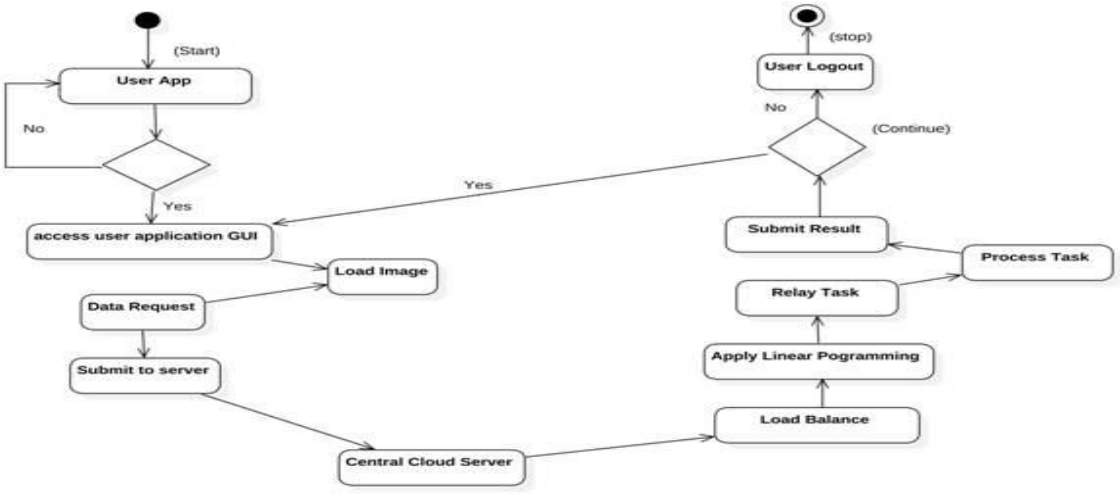


Fig.2:Workflow Diagram

IV. ALGORITHMS

A. Task Scheduling Algorithm:

1) Input: Image as a task [T] for Image processing.:Scheduling of tasks based on capacity of smartphones = T / m (partition of image based on number of smartphones)
2) m: Number of processing nodes. : Number of available tasks in a uniform parallel real time system.(n1,n2…..)
3) Let s1, s2…. sm denote the computing capacity of available processing node
4) Sort sj in a descending order (sj > = sj+1 for all) and let the list be S (S= s1, s2,…..sm
5) T: List of tasks to be performed i.e part of image as a task (T = t1, t2,t3……tm
6) Repea
7) Repea
8) Assign task ti to sj.
9) deduct sj from S
10) deduct ti from T
11) until all ti gets assign
12) Re-sort S based on new addition or deletion of node apacity
13) Re-sort T based on new addition or deletion of tasks
14) . until all the tasks are executed

B. Image Processing Algorithm

1) All grayscale algorithms utilize the same basic three- step process:
2) Get the red, green, and blue values of a pixel Use fancy math to turn those numbers into a single gray value Replace the original red, green, and blue values with the new gray value
3) calculate gray value as Gray = (Red + Green + Blue) / 3
4) Recognize that the actual code to implement such an algorithm looks like:

```
For Each Pixel in Image
Red = Pixel.Red
Green = Pixel.Green
Blue = Pixel.Blue
Gray = (Red + Green + Blue) / 3
assign new gray value to each pixel
Pixel.Red = Gray
Pixel.Green = Gray
Pixel.Blue = Gray
```

C. Mathematical Calculations

Time reuired for processing:

$T = Ej * ni + tj *(ni + cij)$

Where,

T- total time require for processing

Ej Size of task j in KB

ni- time require to take 1KB of data from server to phone

cij- time require to execute task j on 1KB

tj-total count of task

V. EXPERIMENTAL RESULTS

The proposed system is implemented on Linear Programming Platform which smartly divides the task according to volunteers which gives fast processing speed and obtains optimized results. The final results of the proposed system can

be expected as given below –available resources and equal distribution of the load according to the computing capabilities of the resources to ensure that there should not be any under or over utilized resource. So this load balancing framework works as a cloud supporter to reduce its workload. For distributed computing we have used Image processing as a task, but in future this can be used for any other data and task

A. The particulars about platform and technology used:

1) Hardware Requirements

a) i5 Processor.
b) Hard Disk: 40 GB.
c) Floppy Drive: 44 Mb.
d) Ram: 512 Mb(Client),4GB(Server)
e) Mobile phone : ANDROID

2) Software Requirements

a) Operating system: Windows 7 or Windows 8.
b) Coding Language : Java
c) Editor: Netbeans 7.1
d) JDK1.6
e) Database : MySql
f) Webserver: glassfish
g) IDE : Eclipse

B. Testing

The tests were conducted in a 1 Server - 3 Clients environment. The devices are specified in Table I.

TABLE I Specifications and roles of Android Devices used for Testing

Name	RAM	CPU	API	Role
Desktop	4GB	Intel® Core™ i3 CPU 1.70GHz	-	Server
Redmi-4	2GB	Octa-Core Max 1.40GHz	23	Client Volunteer
Moto-G2	2GB	1.2GHz quad-core snapdragon processor	21	Client Volunteer
OPPO	2GB	Qualcomm MSM8916Quad Core	21	Client Volunteer

VI. CONCLUSION

By using this framework based on test conducted it shows time required for different number of devices(volunteers) as shown in following table:

TABLE II Execution time for varying number of devices

Number of Devices	Approximate time in seconds
1	104.20
2	90.54
3	70.34
4	61.54

By using above result sets graphical representation of computing time Vs number of devices is as follows:

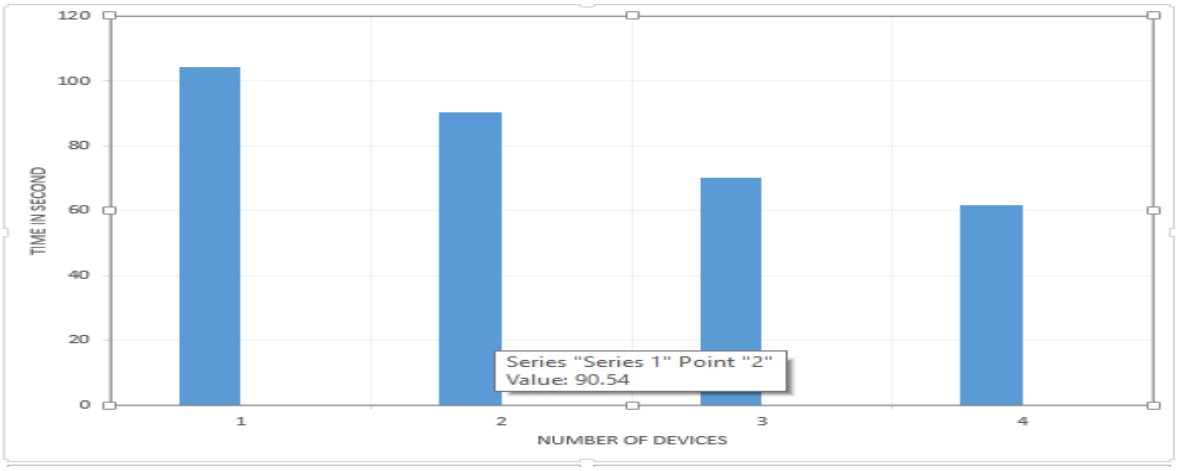


Fig.3 Graphical representation of computing time for varying devices

According to this system, it is found that distributed computing system is the best option to handle the multiple task requests on cloud server. Also to support cloud server for processing these tasks the best and novel idea is using the smart phones as they has good computing capability that can compete with capability of computer. The architecture provides the increasing processing capabilities of smart phone and sizable computing infrastructure. The load balancing framework is necessary to completely utilize all the available resources and equal distribution of the load according to the computing capabilities of the resources to ensure that there should not be any under or over utilized resource. So this load balancing framework works as a cloud supporter to reduce its workload. For distributed computing we have used Image processing as a task, but in future this can be used for any other data and task. Developments are aimed at incorporating all the other mobile Operating Systems such as iOS and support for Windows- based smartphones. Improvements in efficiency can be obtained by implementing runtime load distribution. and by incorporating popular distributed system concepts such as secondary root node for handling root node failure.

REFERENCES

[1] T.Udhaya kumar1, Dr. Radha Senthilkumar2 Department of Information Technology, CWC* - Secured distributed computing using Android devices IEEE 2016.

[2] S. Schildt, F. Busching, E. Jorns, and L. Wolf, CANDIS: Het- erogeneous mobile cloud framework and energy cost-aware schedul- ing in Proc. IEEE Int. August 2013.

[3] Mustafa Y. Arslan, Indrajeet Singh, Shailendra Singh, Har- sha V. Madhyastha, Karthikeyan Sundaresan, Senior Member, IEEE, and Srikanth V. Krishnamurthy, Fellow, IEEE, CWC: A Distributed Computing Infrastructure Using Smartphones, IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 14, NO. 8, August 2015.

[4] D. Datla, H. I. Volos, S. M. Hasan, J. H. Reed, and T. Bose, Taskallocation and scheduling in wireless distributed computing networks, Analog Integr. Circuits Signal Process, vol. 69, nos. 2/3,pp. 341 353, Dec. 2011

[5] F.BeuschingS. Schildt, and L. Wolf, Droid Cluster: Towards smartphone cluster computingThe streets are paved with poten- tial computer clusters, in Proc. 32nd Int. Conf. Distrib. Comput. Syst. Workshop, Jun. 2012, pp. 114117.

[6] S. Ray and A. De Sarkar Execution Analysis Of Load Balancing Algorithms In Cloud Computing Environment In Proc. Interna- tional Journal on Cloud Computing: Services and Architecture (IJCCSA),Vol.2, No.5, October 2012.

[7] S. Sethi, A. Sahu, and S. K. Jena Efficient load Balancing in Cloud Computing using Fuzzy Logic In Proc. IOSR Journal of Engineering (IOSRJEN) ISSN: 2250-3021 Volume 2, Issue 7(July 2012), PP 65-71.

[8] K.Al Nuaimi, N. Mohamed, M. Al Nuaimi and J. Al-Jaroodi A Survey of Load Balancing in Cloud Computing: challenges and Algorithms In Proc. IEEE Second Symposium on Network Cloud Computing and Applications 2012.

[9] C. Wang and Z. Li, A computation offloading scheme on hand- held devices, J. Parallel and Distributed Computing, vol.64, no.6, pp.740-746, June 2004.

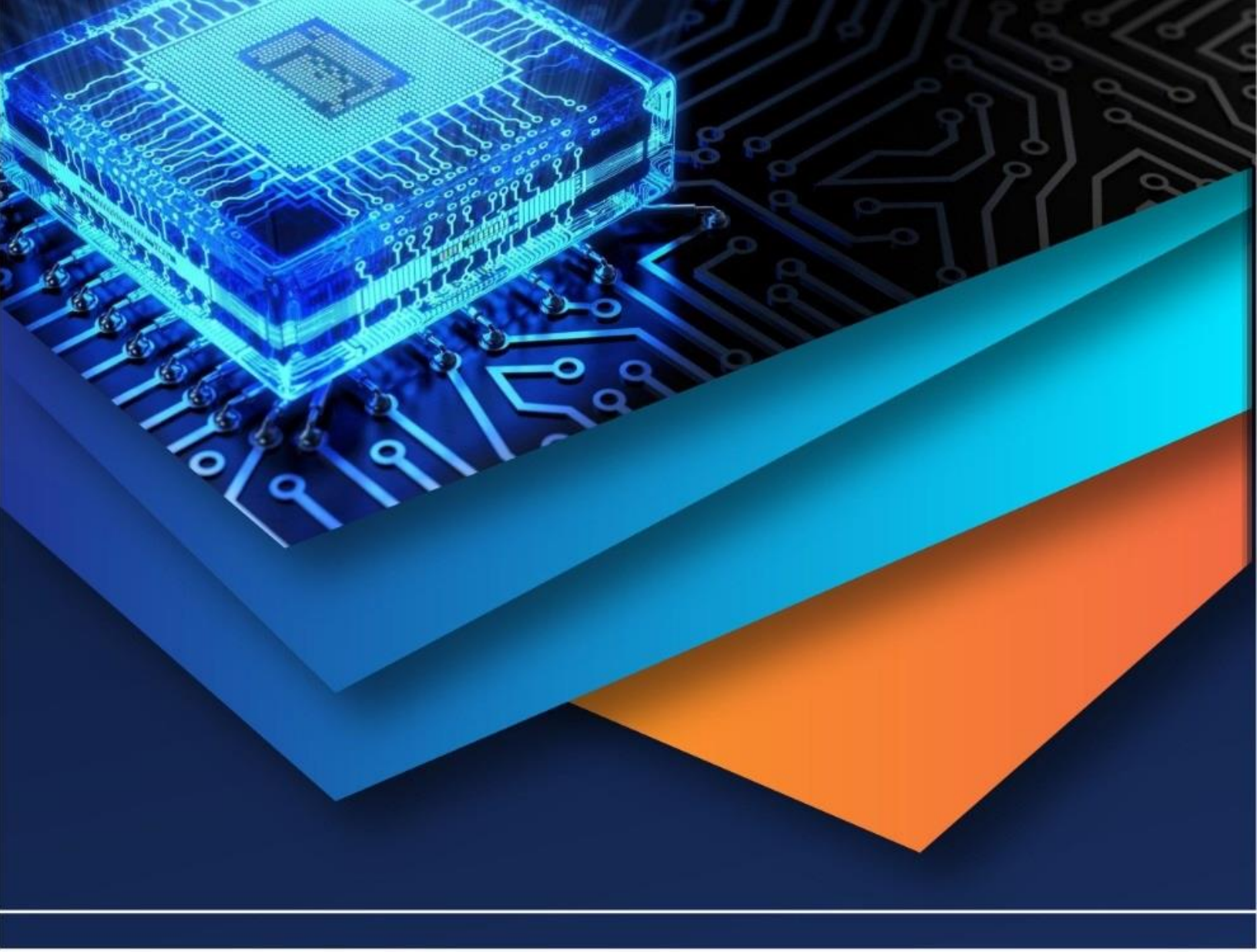
[10] C. Wang and Z. Li, A computation offloading scheme on hand- held devices, J. Parallel and Distributed Computing, vol.64, no.6, pp.740-746, June 2004.

[11] M. Katyal and A. Mishra A Comparative Study of Load Bal- ancing Algorithms in Cloud Computing Environment In Proc. In- ternational Journal of Distributed and Cloud Computing Volume 1 Issue 2 December 2013.

[12] http:/ / books. google. co. in/ books?id=smBw4-xvfrIC lpg=PP1 dq=imagef=false

[13] http:/ / www. imagingbook. com/

[14] http:/ / www. archive. org/ details/ Lectures on Image Processing



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY