Suspect Detection System-An Architecture Based on Surveillance Visual Analytics using IoT

[1]Anirudh Rodda, [2]SugunaMallika S, [3]M Jaiganesh, [4]Mrunalini M

Abstract— *In the present dayscenariothe issue of physical security is of utmost importance in any area. Providing an automated surveillance system to detectsuspicious personnel or activities withina particularvicinity with an alert feature is the need of the hour. The 21st century most prominent technology – Internet of Things (IoT) is being used in developing smart real time security surveillance systems, and providing enhanced performance and effective results by eliminating humansupervision to maximum extent. This new technology is effective both cost and storage wise. Here, the data can be transferred to a remote server such as cloud. Also, the user will be notified via e-mail after an unusual activity/movement is captured. Further, visual analytics help~~s~~ to investigate the data captured and take decisions. This paper is intended to improve the understanding of related tool, technology and methodology used to design and implement such smart surveillance system systematically. In this paper, a methodology is proposed to capture the visitors' image, identify the unknown visitors within the vicinity, storing the unknown visitors'data to the cloud server, processing the unknownvisitors' logs and sending an e-mail notification to the security departmentof the suspicious visitor.*

Keywords: *physical security, automated surveillance system, Visual Analytics*

I. INTRODUCTION

Video surveillance is the process of monitoring an activity and observing for specific behaviors that areeither improper or that may point to the existence ofsuspiciousbehavior. The use of video surveillance systemsincreased significantly in the last several years. It is widely being adopted in public transportation systems, healthcare systems, public safety systems, etc. Often these systems work together as an integrated system to remotely monitor the critical areas to reducecrime~~s~~. The CCTV (Closed-Circuit Television) is one of the basic general video surveillance. CCTV surveillance operates in three modes: active, passive, and archival. From the utility of the currently deployed CCTV surveillance systems, it is observed that they are not fully efficient for either of these described modes of operation [3]. Hence an automated video surveillance technology based on the recent advances in object detection and tracking has been developed. This system is implemented

[1]*Department of Computer Science and Engineering, CVR College of Engineering, Telangana, India*
[2]*Department of Computer Science and Engineering, CVR College of Engineering, Telangana, India*
[3]*Department of Computer Science and Engineering, CVR College of Engineering, Telangana, India*
[4]*Department of Master of Computer Applications, M S Ramaiah Institute of Technology, Bangalore, India.*

Received: 18 Mar 2020 | Revised: 16 Apr 2020 | Accepted: 21 May 2020

asa software that runs on an ordinary desktop computer, which allows efficient summarization and browsing of captured video data [10]. The most important consideration in a video surveillance system is the quality of the image received, and this is largely dependent on the camera and recording equipment used. Also, it is observed that, the surveillance is highly manpower intensive, error-prone and not easily scalable. In addition, enormous amount of video recording has to be stored, of which only a small amount of data may be useful. There is a need to reduce the manual supervision and verification to improve the effectiveness of the video surveillance system. Hence the solution is intelligent video surveillance system.

An intelligent video surveillance system is a networked and distributed IP-based video surveillance system with video analytics algorithms and interface to external sensors. It requires less manual supervision and has the intelligence to provide real-time and automatic event and alarm notifications along with techniques for optimizing storage and network bandwidth and advanced management features on an open architecture. Some of the key concerns include lack of regulation, standards, common operating specifications and performance evaluation. This could result in a compromise of cost, quality of equipment, and the scalability and security against short-term benefits.

Intelligent surveillance system needs the use of both control system and information technologies so that they can be controlled and operated from anywhere with the help of most prominent technology – Internet of Things (IoT) [13]. IoT is being used in developing smart real time security surveillance systems, and providing enhanced performance and effective results by eliminating the humansupervision [16,18]. There are several challenges in implementing and maintaining the intelligent video surveillance systems like: data storage, data access and management, data security etc.

Today, data is produced at an incredible rate and the ability to collect and store the data is increasing at a faster rate than the ability to analyze it. Over the last decade, a large number of automatic data analysis methods have been developed. However, the complex nature of many problems makes it indispensable to include human intelligence at an early stage in the data analysis process. Visual Analytics methods allow decision makers to combine their human flexibility, creativity, and background knowledge with the enormous storage and processing capacities of today’s computers to gain insight into complex problems. Using advanced visual interfaces, humans may directly interact with the data analysis capabilities of today’s computer, allowing them to make well-informed decisions in complex situations.

In thiswork, a methodology is proposed to capture the visitor’s images, identifysuspicious visitors within the vicinity, processing the suspicious visitor’s logs and sending an e-mail notification to the security department. Here, when the visitor enters the premises, the camera captures the images of every visitor. Then the recognition system checks to see if the visitor is known, if known he/she is ignored. If the visitor is unknown, then his/her log files are created and stored in the cloud for further processing. If the unknown visitor is found suspicious, security personnel gets an email alert and can view the surveillance report.

The methodology is implemented to develop a system ‘Suspect Detection System-An Architecture Based on Surveillance Visual Analytics using IoT’, using Raspberry Pi, OpenCV, Amazon Web Services (AWS), Hadoop and Python. Face recognition of each visitor entering the premises is done by the camera using OpenCV and Raspberry Pi. If the visitor is unknown to the premises, a log file is generated and stored in the cloud using Boto3 the AWS SDK for Python.Unknown visitor’s logs stored in the cloud are downloaded to the Hadoop system and are further analyzed using MapReduce Programming model written in Python. Summarize the results and generate a report for each unknown visitor.Send E-mail to the security officer if any visitor is found suspicious. For further investigation the security department can obtain the image of the suspect from the AWS S3 bucket.

The paper is further organized as follows. Existing work related to SDS is summarized in Section II. Proposed system architecture is presented in Section III. The methodology to design and implementthe system 'Suspect Detection System-An Architecture Based on Surveillance Visual Analytics using IoT', is presented in Section IV. In Section V, the methodology illustration and results obtained using the proposed methodology is discussed.The conclusion and future work are presented in Section VI.

II. EXISTING WORK

A sharpincrease in crime on a worldwide scale has expanded the opportunities for utilizing the face recognition technology. Face recognition has become one of the most emerging fields in computer vision. Several biometric systems are being used currently which are way more robust and provide more accuracy e.g. iris and fingerprint scanners, however, in the present time these technologies are better compared to facial recognition but don't show the same scope of improvement[4,5,11].

Y. Cai et al.proposed Advanced driver assistant system (ADAS) that is widely used night time pedestrian detection system, the need for an effective method of identification of suspected pedestrians to prevent the fatal accident that is most likely to occur at low lighted times of the day, especially at nights, is very high [19].

Shao et al. proposed a novel intelligent solution that processes and utilizes video surveillance data based on event identification and alarming messages from front end cameras. The key aspects of this monitoring system are Wandering and Fast movement. The solution is detecting the abnormal events and passing the information through frontend cameras. The abnormal behaviours are identified clearly for each type of event that occurs. Bank robbery, Robbery events have abnormal behaviours as wandering and fast movement [3,14].

P. Merla et al.using the Hadoop MapReduce framework on the AWS cloud platform performed data analysis on YouTube data. Apache Hadoop, a data processing framework can analyse large data sets with very high speed. The proposed system processes real-time YouTube data analysed using the MapReduce algorithm to get Top rated categories, uploads, and most viewed videos. The data is extracted using YouTube's Reporting and Analytics APIs and stored as a CSV file in the Hadoop Distributed File System (HDFS). The MapReduce retrieves data from HDFS to perform data processing and summaries the analytic data and stores the results back into HDFS. The obtained results are represented on a graphical user interface [12].

Due to the shortcomings in the video surveillance systems such as poor image quality, storage space, prices, there is a need for real time security surveillance system.Naga et al proposed a system design that uses Motion Detection algorithm being implemented on low processing power chip Raspberry pi 2 and Pi camera written in Python. This significantly decreases the storage usage and also save investment cost. The algorithm enables live video streaming with detection of moving objects and get alarm when motion is detected and sends photos, videos to a cloud server directly using pi camera [17].

In Big data era, storing and processing huge surveillance video data poses challenges. A novel intelligent video surveillance system with massive surveillance video data storage and processing with intelligent video analytics techniques is proposed by Zhenfeng Shaoet al. This system includes the intelligent pre-alarming for security risk events, smart storage for recorded video and rapid retrieval associated with specific suspects [20].

III. SYSTEM ARCHITECTURE

The Architecture of the system *'Suspect Detection System-An Architecture Based on Surveillance Visual Analytics using IoT',* is shown in Figure 1. From Figure1, it is observed that it is a three-Tier architecture. Tier-1 includescamera and Raspberry Pi), Tier-2 includes data processing using HadoopSystem, Tier-3 includes cloud storage i.e., AWS Simple Storage Service (S3).

The cameras in this system are considered as Agents that captures the activity of every visitor to the premises. The face recognition model identifies the unknown visitor and their log files are created and stored in the AWS S3 bucket. Further, the Hadoop system downloads these log files from S3 bucket. The Hadoop system performs data processing on these log files using MapReduce programming model, summarizes the results and generates a report based on summary results. The user interface system sends an email notification to the security officer if any unknown visitor is found suspicious. The security department can access the AWS S3 bucket for an image of the suspect for the further investigation process.

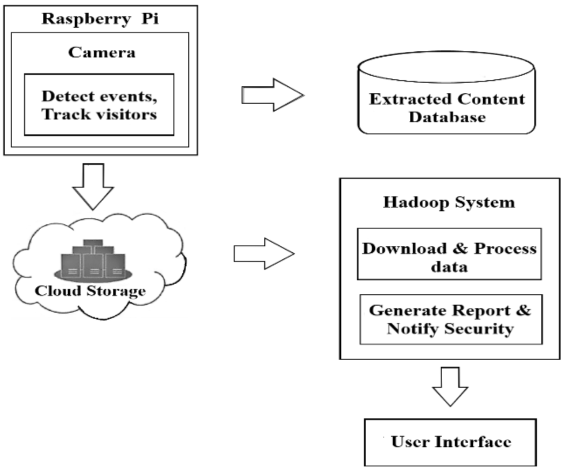

Raspberry Pi
Camera
Detect events, Track visitors
Extracted Content Database
Cloud Storage
Hadoop System
Download & Process data
Generate Report & Notify Security
User Interface

Figure 1. System Architecture

IV. METHODOLOGY

The proposed methodology helps to design and implementthe system'Suspect Detection System-An Architecture Based on Surveillance Visual Analytics using IoT', which provide security to authorized premises and help the security department to keep the premises away from intrusions.Various phases involved in executing the methodology are provided in Table 1.

Table1. Suspect Detection System-An Architecture Based on Surveillance Visual Analytics using IoT-Methodology

Step1. Face Detection: - Face identification isdone for each visitor entering the premises by the camera.

Step2. Monitoring the visitor’s activities: - The visitor’s activities will be observed by the camera.

Step3. Storing the visitor’s data: - If the visitor is unknown to the premises, then a log file is generated and stored in the cloud.

Step4. Downloading the visitor’s data: -Unknown visitor’s logs which are stored in the cloud are downloaded to the Hadoop system for further processing.

Step5. Analyzing the data: - Analyze the data using the Hadoop MapReduce.

Step6. Tracking the visitor visits: - The visitor’s number of visits within the premises is analyzed using the Hadoop MapReduce.

Step7. Summarizing the results: - Summarize the results obtained from the Hadoop MapReduce Output.

Step8. Generating the report: - Based on the summary results, generate a report for each unknown visitor.

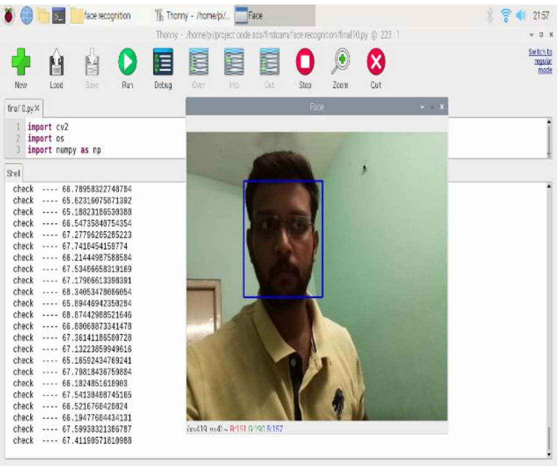
Step9. E-mail Notification: - Send E-mail to the security officer if the visitor is found suspicious.

V. METHODOLOGY IMPLEMENTATION, RESULTS AND DISCUSSION

- ***Face Detection***

OpenCV module should be installed in the raspberry pi to perform face detection as shown in Figure 2. In facial detection, the first step is to detect the faces as an input image. Face detection uses OpenCV’s cascade classifier called Haar Cascades which uses the Ada Boost algorithm to detect multiple facial features. When an object is detected it is converted to a grayscale image. The cascade classifier decides whether the detected image is a human or not if it detects a human face then it examines the facial features and prints a rectangular frame on the detected face. A facial recognition system utilizes biometrics to map the facial features from the input image. It compares the input image data with the database of known images to find a match. Each captured image is given a unique ID and these captured images are stored in a dataset. These stored images are trained using OpenCV specific functions and results are stored in a .yml file.

From Figure 2 it is observed that, in the proposed system, the camera does the face identification of each visitor entering the premises. Camera checks to see if the visitor is a known person to the premises.If the visitor is known then visitor activities are not recorded.



```
import cv2
import os
import numpy as np
```

```
check  ---- 66.78958322748784
check  ---- 65.62316075871392
check  ---- 65.18823186530388
check  ---- 66.54735840754354
check  ---- 67.27796205285223
check  ---- 67.7410454159774
check  ---- 66.21444987588584
check  ---- 67.53406658319169
check  ---- 67.17906613398391
check  ---- 68.34053478086054
check  ---- 65.89446942350284
check  ---- 68.87442988521646
check  ---- 66.88068873341478
check  ---- 67.36141186580728
check  ---- 67.13223859949616
check  ---- 65.16592434769241
check  ---- 67.79818436759884
check  ---- 66.1824851618903
check  ---- 67.54138488745165
check  ---- 66.5216768428824
check  ---- 66.19477684434131
check  ---- 67.59930321386787
check  ---- 67.41190571810988
```

Figure 2. Face Detection & Image Capturing

- ***Monitoring the visitor's activities***

If the visitor is unknown then the visitor's activities must be recorded. The unknown visitor is given a unique ID and then a log file is generated for individual unknown visitor as shown* in Figure 3.

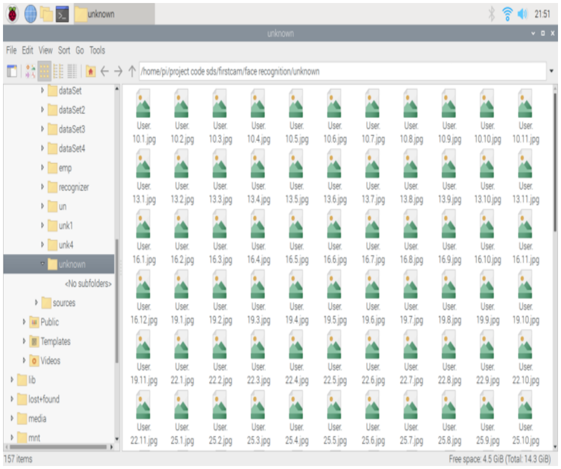
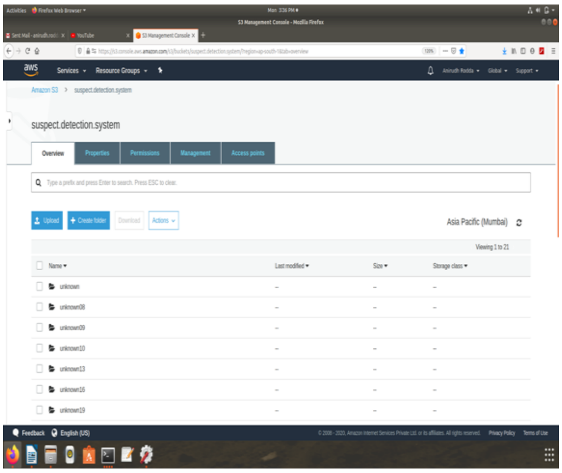


Figure 3. Local Data Store with Unknown Visitor Images

Figure 3 depicts the system after recognizing the visitor as unknown with a set of images of the unknown visitors trained and stored in a local folder *unknown*.

- ***Storingthe data to the cloud***

Every movement of the unknown visitor is observed. If the visitor is often visiting the prohibited area, then a log file is generated and stored in the cloud as shown in Figure 4.



suspect.detection.system

Viewing 1 to 21

Name	Last modified	Size	Storage class
unknown	–	–	–
unknown08	–	–	–
unknown09	–	–	–
unknown10	–	–	–
unknown13	–	–	–
unknown16	–	–	–
unknown19	–	–	–

Figure 4 Log File of Every Unknown Visitor in AWS S3 Bucket

From Figure 4 it is observed that, AWS S3 bucket*suspect.detection.system*, is created andeach folderin the bucket is assigned a name as unknownfollowed by a unique Id representing the unknown visitor.

An image of the unknown visitor is saved in their respective folder. A text file with a label as the current date is created and every movement of the visitor is saved in that text file as shown in Figure 5.

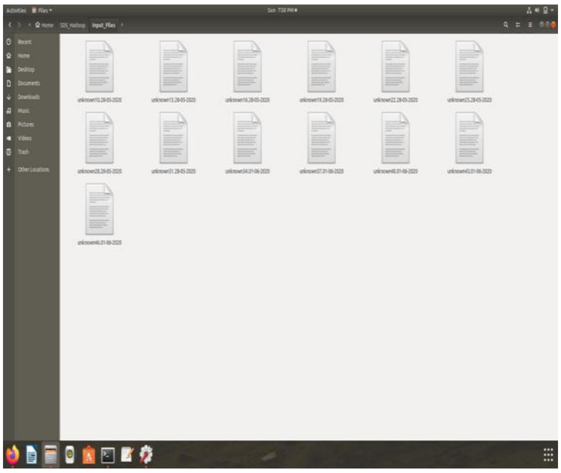


Figure 5. Unknown Visitor Log File

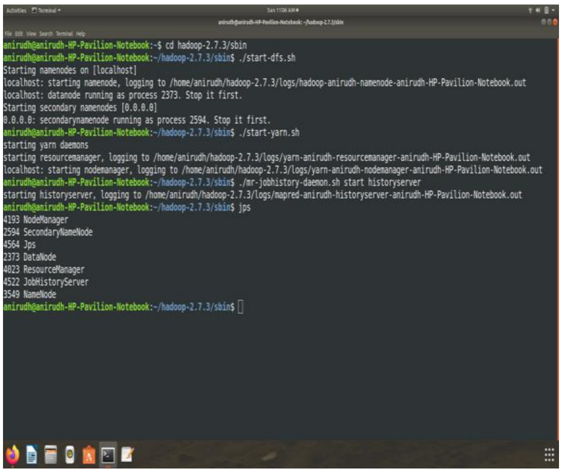
Figure 5 shows the log files of an unknown visitor. These log files contain a text file with a label as current date with area name, camera number as contents in it, and an image of the visitor with the label of the image as the folder name.

- ***Retrievingthe data from the cloud***

The unknown visitors log data from AWS S3 bucket will be download to the Hadoop system for further processing to identify whether the unknown visitor is a suspect or not as follows.

Start the Hadoop services using start-all.sh command or can individually run the services using start-dfs.sh and start-yarn.sh commands as shown in Figure 6.

Received: 18 Mar 2020 | Revised: 16 Apr 2020 | Accepted: 21 May 2020

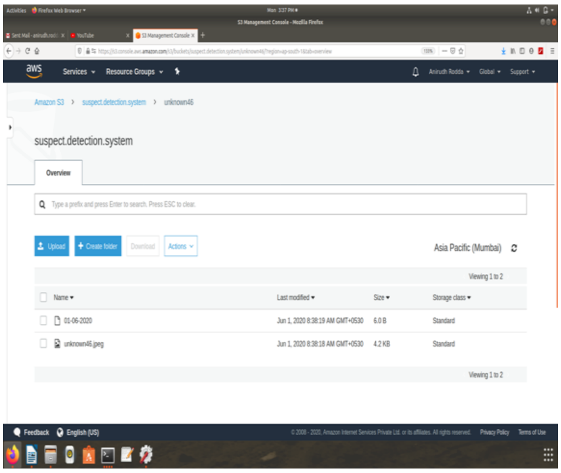


Activities
Terminal
anirudh@anirudh-HP-Pavilion-Notebook: ~/hadoop-2.7.3/sbin
File Edit View Search Terminal Help

```
anirudh@anirudh-HP-Pavilion-Notebook:~$ cd hadoop-2.7.3/sbin
anirudh@anirudh-HP-Pavilion-Notebook:~/hadoop-2.7.3/sbin$ ./start-dfs.sh
Starting namenodes on [localhost]
localhost: starting namenode, logging to /home/anirudh/hadoop-2.7.3/logs/hadoop-anirudh-namenode-anirudh-HP-Pavilion-Notebook.out
localhost: datanode running as process 2373. Stop it first.
Starting secondary namenodes [0.0.0.0]
0.0.0.0: secondarynamenode running as process 2594. Stop it first.
anirudh@anirudh-HP-Pavilion-Notebook:~/hadoop-2.7.3/sbin$ ./start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/anirudh/hadoop-2.7.3/logs/yarn-anirudh-resourcemanager-anirudh-HP-Pavilion-Notebook.out
localhost: starting nodemanager, logging to /home/anirudh/hadoop-2.7.3/logs/yarn-anirudh-nodemanager-anirudh-HP-Pavilion-Notebook.out
anirudh@anirudh-HP-Pavilion-Notebook:~/hadoop-2.7.3/sbin$ ./mr-jobhistory-daemon.sh start historyserver
starting historyserver, logging to /home/anirudh/hadoop-2.7.3/logs/mapred-anirudh-historyserver-anirudh-HP-Pavilion-Notebook.out
anirudh@anirudh-HP-Pavilion-Notebook:~/hadoop-2.7.3/sbin$ jps
4193 NodeManager
2594 SecondaryNameNode
4564 Jps
2373 DataNode
4023 ResourceManager
4522 JobHistoryServer
3549 NameNode
anirudh@anirudh-HP-Pavilion-Notebook:~/hadoop-2.7.3/sbin$ 
```

Figure 6. Start Hadoop Services

Download the log file of every unknown visitor from the AWS S3 bucket and store them in the *input* folder of Hadoop local system as shown in Figure 7



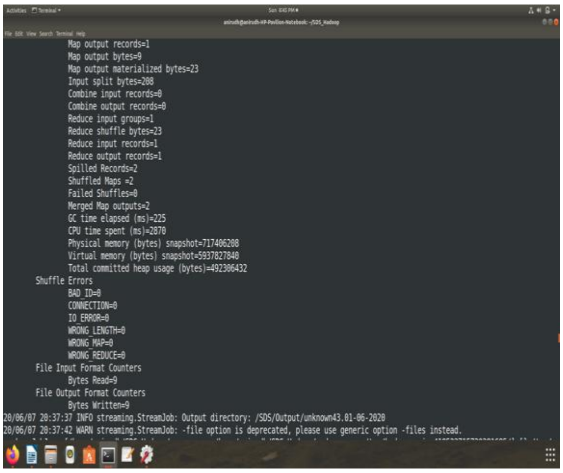
Name	Last modified	Size	Storage class
01-06-2020	Jun 1, 2020 8:38:19 AM GMT+0530	6.0 B	Standard
unknown46.jpeg	Jun 1, 2020 8:38:18 AM GMT+0530	4.2 KB	Standard

Figure 7. Unknown Visitor Files in Input Folder

- ***Analyzing the data***

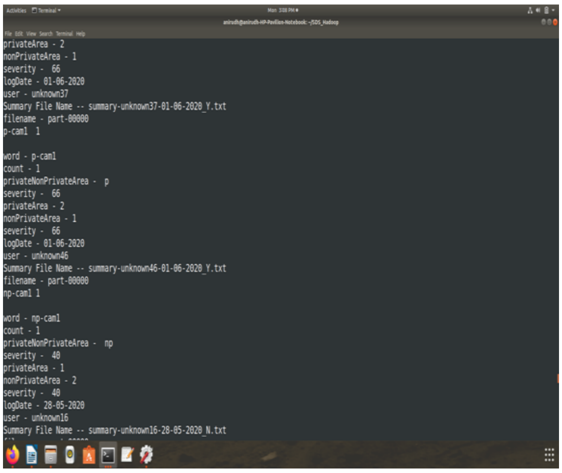
The log files retrieved from the cloud i.e., AWS S3 bucket are taken as input by the Hadoop system. The downloaded log files are copied from the local system to the Hadoop Distributed File System (HDFS) which are taken as input to Hadoop MapReduce jobs. The Hadoop system runs MapReduce jobs and analyses all the log files as shown in Figure 8 and Figure 9.

Received: 18 Mar 2020 | Revised: 16 Apr 2020 | Accepted: 21 May 2020


Activities
Terminal
Sun 6:45 PM
anirudh@anirudh-HP-Pavilion-Notebook: ~/SDS_Hadoop
File Edit View Search Terminal Help

```
                Map output records=1
                Map output bytes=9
                Map output materialized bytes=23
                Input split bytes=208
                Combine input records=0
                Combine output records=0
                Reduce input groups=1
                Reduce shuffle bytes=23
                Reduce input records=1
                Reduce output records=1
                Spilled Records=2
                Shuffled Maps =2
                Failed Shuffles=0
                Merged Map outputs=2
                GC time elapsed (ms)=225
                CPU time spent (ms)=2870
                Physical memory (bytes) snapshot=717406208
                Virtual memory (bytes) snapshot=5937827840
                Total committed heap usage (bytes)=492306432
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=9
        File Output Format Counters
                Bytes Written=9
20/06/07 20:37:37 INFO streaming.StreamJob: Output directory: /SDS/Output/unknown43.01-06-2020
20/06/07 20:37:42 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
```

Figure 8. MapReduce Job Completion Status


Activities
Terminal
Mon 3:08 PM
anirudh@anirudh-HP-Pavilion-Notebook: ~/SDS_Hadoop
File Edit View Search Terminal Help

```
privateArea - 2
nonPrivateArea - 1
severity -  66
logDate - 01-06-2020
user - unknown37
Summary File Name -- summary-unknown37-01-06-2020_Y.txt
filename - part-00000
p-cam1  1

word - p-cam1
count - 1
privateNonPrivateArea -  p
severity -  66
privateArea - 2
nonPrivateArea - 1
severity -  66
logDate - 01-06-2020
user - unknown46
Summary File Name -- summary-unknown46-01-06-2020_Y.txt
filename - part-00000
np-cam1 1

word - np-cam1
count - 1
privateNonPrivateArea -  np
severity -  40
privateArea - 1
nonPrivateArea - 2
severity -  40
logDate - 28-05-2020
user - unknown16
Summary File Name -- summary-unknown16-28-05-2020_N.txt
```

Figure 9. MapReduce Job Output with Severity Count

- ***Summarizing the results***

The Hadoop system after performing the MapReduce job on each log file gives the output as the number of visits of all the unknown visitor's within the premises. The unknown visitor's visit count as taken as input and generates a severity value for each visitor's visits. If the visits of the unknown visitor are beyond the threshold value then the particular unknown visitor is confirmed to be a suspect.Figure 10 shows the contents of the summary folder, which contain unknown visitor Id, total visits of the unknown visitor, severity count of the unknown visitor, log date of the unknown visitor, and camera number under which the unknown visitor was captured. The filename has a flag value Y or N to identify that filename with Y as a flag value indicate it as a suspect.

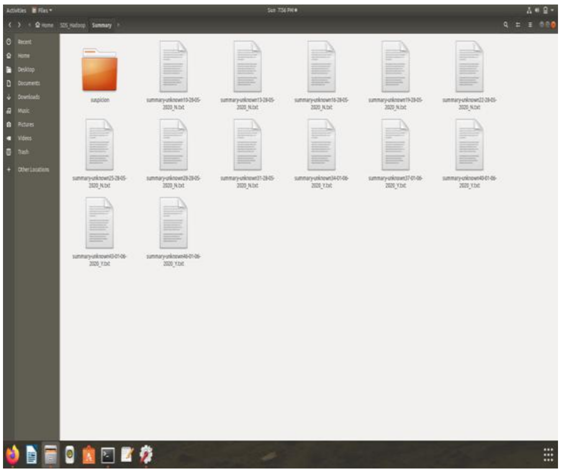
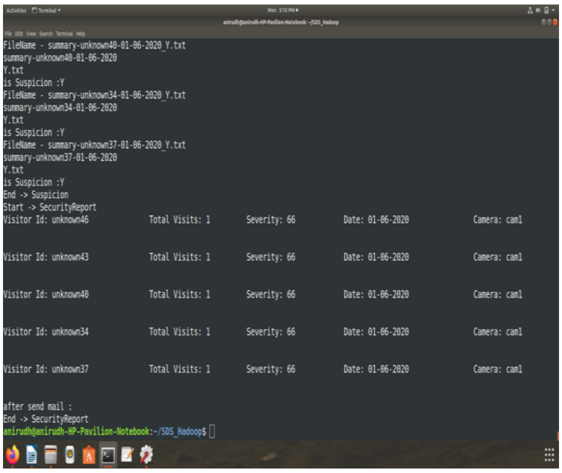

Activities
Files
Sun 7:56 PM
Home
SDS_Hadoop
Summary
Recent
Home
Desktop
Documents
Downloads
Music
Pictures
Videos
Trash
Other Locations
suspicion
summary-unknown10-28-05-2020_N.txt
summary-unknown13-28-05-2020_N.txt
summary-unknown16-28-05-2020_N.txt
summary-unknown19-28-05-2020_N.txt
summary-unknown22-28-05-2020_N.txt
summary-unknown25-28-05-2020_N.txt
summary-unknown28-28-05-2020_N.txt
summary-unknown31-28-05-2020_N.txt
summary-unknown34-01-06-2020_Y.txt
summary-unknown37-01-06-2020_Y.txt
summary-unknown40-01-06-2020_Y.txt
summary-unknown43-01-06-2020_Y.txt
summary-unknown46-01-06-2020_Y.txt

Figure 10. Summary Folder

- ***Report Generation***

A summarized report is generated for every individual suspected visitor which contains visitor's ID, visit count, date of appearance, severity count, and camera number as shown in Figure 11.Further, these contents will be sent in an E-mail to the Security officer.



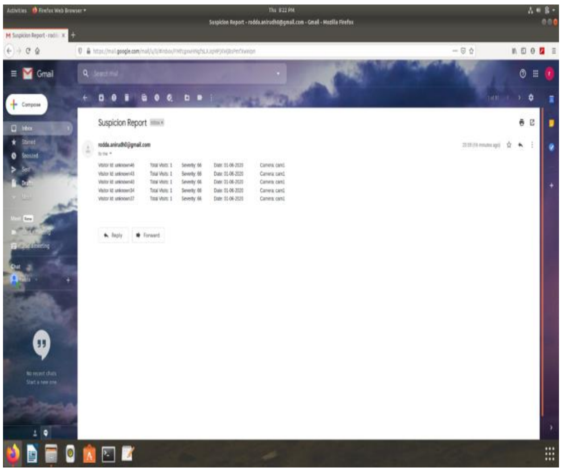
```
FileName - summary-unknown40-01-06-2020_Y.txt
summary-unknown40-01-06-2020
Y.txt
is Suspicion :Y
FileName - summary-unknown34-01-06-2020_Y.txt
summary-unknown34-01-06-2020
Y.txt
is Suspicion :Y
FileName - summary-unknown37-01-06-2020_Y.txt
summary-unknown37-01-06-2020
Y.txt
is Suspicion :Y
End -> Suspicion
Start -> SecurityReport
Visitor Id: unknown46           Total Visits: 1       Severity: 66        Date: 01-06-2020          Camera: cam1


Visitor Id: unknown43           Total Visits: 1       Severity: 66        Date: 01-06-2020          Camera: cam1


Visitor Id: unknown40           Total Visits: 1       Severity: 66        Date: 01-06-2020          Camera: cam1


Visitor Id: unknown34           Total Visits: 1       Severity: 66        Date: 01-06-2020          Camera: cam1


Visitor Id: unknown37           Total Visits: 1       Severity: 66        Date: 01-06-2020          Camera: cam1


after send mail :
End -> SecurityReport
anirudh@anirudh-HP-Pavilion-Notebook:~/SDS_Hadoop$ 
```

Figure 11. Summary Report

- ***Sending E-mail notification***

The system sends an e-mail notification to the registered E-mail address when the unknown visitor is found suspicious. The E-mail will have the visitor's ID, visit count, date of appearance, severity count, and camera number as shown in Figure 12.

Received: 18 Mar 2020 | Revised: 16 Apr 2020 | Accepted: 21 May 2020



Suspicion Report

rodda.anirudh0@gmail.com

```
Visitor Id: unknown46    Total Visits: 1    Severity: 66    Date: 01-06-2020    Camera: cam1
Visitor Id: unknown43    Total Visits: 1    Severity: 66    Date: 01-06-2020    Camera: cam1
Visitor Id: unknown40    Total Visits: 1    Severity: 66    Date: 01-06-2020    Camera: cam1
Visitor Id: unknown34    Total Visits: 1    Severity: 66    Date: 01-06-2020    Camera: cam1
Visitor Id: unknown37    Total Visits: 1    Severity: 66    Date: 01-06-2020    Camera: cam1
```

Figure 12. E-mail Report

VI. CONCLUSION AND FUTURE DIRECTIONS

In this paper, a methodology is proposed to capture and process the unknown visitor’s log within a vicinity and further sending E-mail notification to the security department if the unknown visitor is found suspicious.The methodology is implemented using the modern tools and technologies for safeguarding various business sections and government organizations which in turn reduces the usage of manpower. That is, there is less human interaction required for the proper functioning of the system as the system is completely automated to safeguard the premises. In future, this methodology can be integrated into the government ‘*Aadhar’*database to get visitor's details.

REFERENCES

1. André F. M. Batista, Pedro L. P. Correa, GiriPalanisamy,”Visual Analytics Improving Data Understandability in IoT Projects: An Overview of the U. S. DOE ARM Program Data Science Tools”, In Proceedings of the 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), IEEE, ISBN: 978-1-5090-2834-4, 2016.
2. Bowen Du ,Chuanren Liu, Wenjun Zhou , Zhenshan Hou, and Hui Xiong, “Detecting Pickpocket Suspects from Large-Scale Public Transit Records”, IEEE Transactions on Knowledge And Data Engineering, Vol. 31, NO. 3, March 2019.
3. Dmitry O. Gorodnichy and Tony Mungham, “Automated video surveillance: challenges and solutions. ACE Surveillance (Annotated Critical Evidence) case study”, NATO SET-125 Symposium "Sensor and Technology for Defence against Terrorism", Mainheim, April 2008.
4. Fahad Parvez Mahdi, Md. Mahmudul Habib, Md. Atiqur Rahman Ahad, Susan Mckeever, A.S.M.Moslehuddin and Pandian Vasant, “Face recognition-based real- time system for surveillance”, Intelligent Decision Technologies, ISSN 1872-4981/17,79-92, 2017.

Received: 18 Mar 2020 | Revised: 16 Apr 2020 | Accepted: 21 May 2020

5. Farah Deeba, Hira Memon, Fayaza Ali Dharejo, Aftab Ahmed, Abddul Ghaffar, "LBPH-based Enhanced Real-Time Face Recognition", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 5, 2019.
6. Gaocheng Liu, Shuai Liu, Khan Muhammad, Arun Kumar Sangaiah, And Faiyaz Doctor, "Object Tracking in Vary Lighting Conditions for Fog Based Intelligent Surveillance of Public Spaces", Special Section on Real-Time Edge Analytics For Big Data In Internet of Things, Vol.6, 10.1109/ACCESS.2018.2834916.
7. Guo-Dao Sun, Ying-Cai Wu, Rong-Hua Liang and Shi-Xia Liu ," A Survey of Visual Analytics Techniques and Applications" State-of-the-Art Research and Future Challenges, Journal of Computer Science and Technology, volume 28, pg: 852–867(2013).
8. https://pythonprogramming.net/raspberry-pi-camera-opencv-face-detection-tutorial/
9. IndrajitPatil , Saurabh Jaiswal , Pallavi Sakhare , Mohammad Shoaib , Asst. Prof. Poonam Gupta, "A Survey on IOT Based Security System", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 5, Issue 11, November 2016
10. Jing Wang ;Zhijie Xu , "Crowd anomaly detection for automated video surveillance", In Proceedings of the 6th International Conference on Imaging for Crime Prevention and Detection (ICDP-15), ISBN: 978-1-78561-131-5, 2015.
11. JoshilaGrace.L.K, K.Reshmi, "Face Recognition in Surveillance System", IEEE Sponsored 2nd International Conference on Innovations in Information, Embedded, and Communication Systems (lCIIECS), 20I5.
12. Konstantin Shvachko, HairongKuang, Sanjay Radia, Robert Chansler, "The Hadoop Distributed File System", in Proceedings of IEEE Symposium. Mass Storage Syst. Technol., 2010, pp. 1-10.
13. P.P.Ray, "A survey on Internet of Things architectures", Journal of King Saud University - Computer and Information Sciences, Vol.30, Issue 3, pg: 291-319 (2018).
14. Pawan Kumar Mishra, G. P. Saroha, "A Study on Video Surveillance System for Object Detection and Tracking", In Proceedings of the International Conference on Computing for Sustainable Global Development, ISBN 978-93-80544-20-5, IEEE, 2016.
15. Prof. A. M. Jagtap, Mr. VrushabhKangale, Mr. Kushal Unune, Mr. PrathmeshGosavi, "A Study of LBPH, Eigenface, Fisherface and Haar-like features for Face recognition using OpenCV", International Conference on Intelligent Sustainable Systems (ICISS 2019) IEEE Xplore Part Number: CFP19M19-ART; ISBN: 978-1-5386-7799-5.
16. Rickin Patel, Vipul K. Dabhi, Harshadkumar B. Prajapati, "A survey on IoT based road traffic surveillance and accident detection system (A smart way to handle traffic and concerned problems)", In Proceedings of the Innovations in Power and Advanced Computing Technologies (i-PACT), ISBN: 978-1-5090-5683-5, IEEE, 2017.
17. S. Naga Jyothi and K. Vijaya Vardhan, "Design and implementation of real time security surveillance system using IoT", In Proceedings of the International Conference on Communication and Electronics Systems (ICCES), ISBN: 978-1-5090-1067-7, IEEE, 2016.

18. SharminAkter, Rehana AfrozSima, Md. Sohid Ullah, Syed Akhter Hossain, “Smart Security Surveillance using IoT”, In Proceedings of the 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), ISBN: 978-1-5386-4693-9, IEEE, 2018.
19. Yingfeng Cai, Ze Liu, Hai Wang, Xiaoqiang Sun, “Saliency-Based Pedestrian Detection in Far Infrared Images”, Vol.5, 10.1109/ACCESS.2017.2695721.
20. Zhenfeng Shao, Jiajun Cai, and Zhongyuan Wang, “Smart Monitoring Cameras Driven Intelligent Processing to Big Surveillance Video Data”, IEEE Transactions on Big Data, Vol. 4, NO. 1, January-March 2018