International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 03 | Mar-2018 www.irjet.net p-ISSN: 2395-0072

Storage Optimization of Video Surveillance from CCTV Camera

D.Suganya[1], R.Shyla[2], V.Jayasudha[3], S. Marirajan[4]

[1,2,3,4] *Department of Electronics and Communication Engineering, Valliammai Engineering College, Kattankulathur-603203, India*

Abstract - *CCTV (Closed Circuit Television) or video surveillance is a useful technology that is mainly used for security purposes. It can be found at many places from public to private locations. The video footage captured by the cameras are stored in secondary devices like pen drives, hard disk drives. The most challenging problem in CCTV camera is storage space occupied by the footage. Because each day, the camera captured a large amount of data and stored it. But some of these data may be useless when there is no activity is performed. Hence compression techniques are used to reduce the storage space. We are proposing an idea to optimize the storage space by reducing the redundant frames. By using MSE (mean squared error) the adjacent frames are compared and the redundant frames are deleted between the adjacent frames of the video.*

Keywords: CCTV, redundant frames, storage optimization, MSE.

I. INTRODUCTION

CCTV camera are basic requirements for security of any places. After certain period of time, the memory space are upgraded. It leads to high maintenance cost. Hence the video size are optimized by removing unwanted video frames. Our project is to design and implement an intelligent system that accepts input as a video sequences. The video sequences are subdivided into frames. The adjacent frames are compared then the will stored frames that are not similar to each other. Footage is mainly stored in secondary storages. So to reduce storage space, compression techniques are used.

A CCTV camera generated a video at its lowest quality is generally recorded at 352X240 resolution and 30fps occupies 10 GBs of storage in 1 day. There are 245 million CCTV cameras in the world right running 24X7 capturing every second of year. Hence storing large amount of data is a huge problem and some cameras automatically delete all the footage after certain period of time.

This approach will optimize the storage maintaining information as well as quality. Whenever the motion sensor detect an image then the camera will turn on, take image and recognize by using open CV technique (Image processing)

Image processing is basically a method of converting an image digitally and performing certain operations on it where the input is an image and output can be an image or functions associated with the image.

II. RELATED WORKS

According to survey on Video Surveillance and Storage by IHS technologies [3] in September 2014, there are 245 million CCTV cameras installed all over the world in 2014. Out of which 71% are from Asia. A CCTV camera generates a video at its low quality is generally recorded at 352X240 resolution and 30 fps occupies 10 GBs of storage in 1 day which leads to 1TB of storage in 138 days. In 2012, Seagate [2] submitted a Technical Review Paper, where compression techniques are done to store the CCTV footage which results in loss of quality and does not affect much on size. So, they suggested to use Backup plus option by Seagate. In closed circuit camera the use of heterogeneous training data and more data is explored to improve its deduction rate. Besides it is proposed to use the transformation parameters of the object space to automatically model and predict the evolution of intrinsic parameter of the camera. Therefore to adjust the detector for better performance [1]. In recent researches done on this topic, cloud storage is suggested for CCTV footage. But it requires fast internet connection in every CCTV cameras.

III. EXISTING SYSTEMS

A technical research paper "Video Surveillance Storage: How Much Is Enough?" in 2012 by Seagate shows compression techniques to store the CCTV footage which results in loss of quality and does not affect much on size of video. So, they suggested to use Backup plus option by Seagate.

And other research done in CCTV includes primary and secondary storage systems. Primary includes the storage device connected to camera where the data is compressed and stored in secondary where secondary is used for backup which is server or cloud. Then the primary is cleaned and indexed to secondary storage. In this technique, data is not lost as the CCTV cameras are used for security purposes. In recent researches done on this topic, they suggested to store the CCTV footage in cloud. But this requires fast internet connection for storing every footage. So, it is not feasible in India in coming years.

IV. PROPOSED METHOD

The main objective is to propose a method to optimize the storage maintaining information as well as quality. Whenever the motion sensor detect an image, then camera will turn on and take image which is recognized

by using open CV technique (Image Processing). In this work, Python is being used as platform for implementing various image processing and computer vision techniques. The video sequences are divided into frames, the adjacent frames are compared. The system will store the frames that are not similar to the adjacent frames.

V. METHODOLOGY

An Incremental Development Methodology is followed in this project. We are considered a multiple development cycles, which makes the life cycle a "multi-waterfall" cycle. Cycles are subdivided into smaller modules which can be easily managed. Each module undergoes requirements, design, implementation and testing phases.

The advantage of this proposed methodology is that we can able to use the software or the program during its build phase so we could improve the design as well as alter the source code at any stage without the need of performing each step individually.

This model is less costly and more flexible to change its scope and requirements. During a smaller iteration, it is very easy to test and debug.

VI. MODULES

The project consists of 3 modules:

1. Extracting frames from video sequences
2. Comparing adjacent frames of video sequences
   a) Comparing MSE of adjacent frames
   b) Deleting the frames having MSE < Threshold
3. Combining the non- similar frames into video sequence

VII. BLOCK DIAGRAM

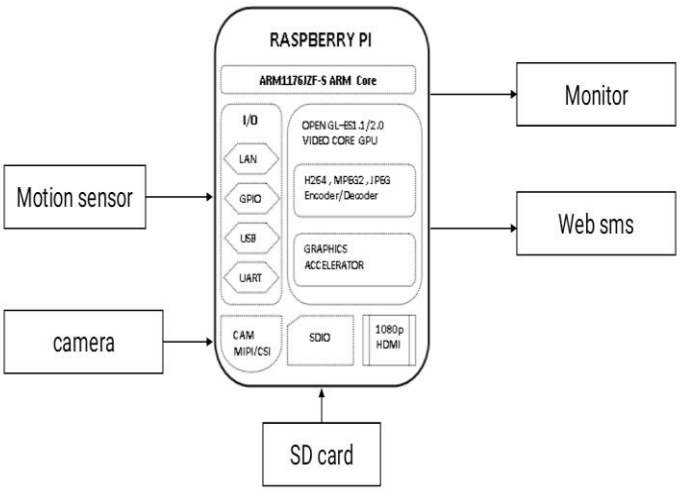

Fig 1. Block diagram of CCTV Camera

VIII. PIN DIAGRAM

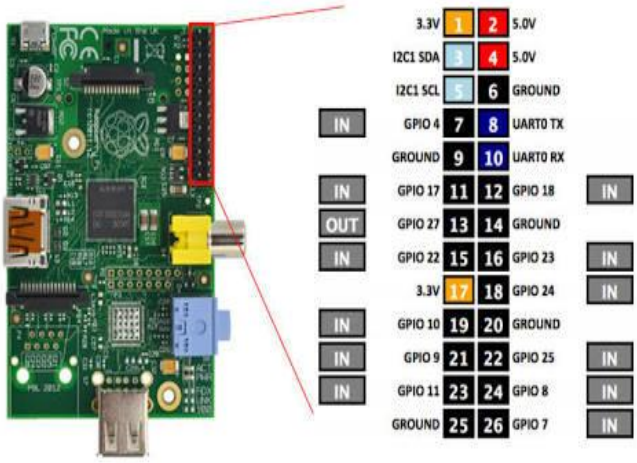

Fig.2 Pin diagram of raspberry pi

IX. FLOW CHART

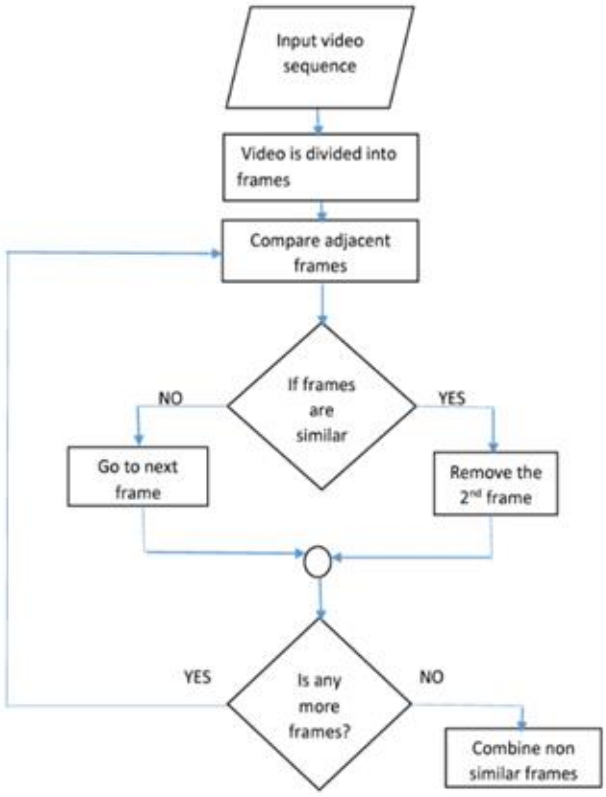

Fig 3. Flow chart of the video recording

X. PROPOSED ALGORITHM

Step 1: Input CCTV video sequences

Step 2: Extract frames from video sequences

Step 3: Calculate the MSE value for each key frame and test frame

Step 4: Set key frame = 1st frame and test frame = 2nd frame

Step 5: if MSE >= Threshold, then key frame = test frame test frame = key frame + 1 save the key frame

else

test frame = test frame + 1 Step 6: if there are more frames

Repeat Step 5

else

Combine the resulted frame to get optimized video with same fps

Delete original video and frames of both videos

XI. ANALYSIS OF EACH MODULE

Module 1 : Extracting frames from Video Sequences

For extracting frame from a video sequence, video is paused at every frame using cv2.waitKey() and that frame is saved as an image cv2.imwrite()

```
import cv2
vc=cv2.VideoCapture('D:\\PROJECT\\
test.mp4')
c=1
if vc.isOpened():
rval , frame = vc.read()
else:
rval=False
print("error")
while rval:
rval, frame = vc.read()
cv2.imwrite("D:\\frames\\"
+ str(c) + '.jpg',frame)
c = c + 1
cv2.waitKey(1)
vc.release()
```

Module 2: Comparing adjacent frames of Video sequences

MSE of each frame is calculated and then the threshold is predicted through recursive testing for the footage in the environment. The threshold value we are using here is to be 5.

The MSE can be calculated by,

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$$

Where,

m - no. of horizontal pixels (width of frame)

n - no. of vertical pixels (length of frame)

I, K - frame

i(i, j) - pixel intensity value at the co-ordinate (i,j)

If MSE = 0, then the frames are 100% same and if MSE = 2, the difference between frames are not visible to human eye. And in our video there is a timestamp, so after many test cases, MSE turn out to be 5.

Threshold is judged through recursive testing for the footage and the threshold for the video we used comes out to be 5. And then the frame is taken as a key and MSE is calculated of that frame and its adjacent frame named test frame, if the MSE is less than 5, then the test frame is discarded and if MSE is greater than 5 then the test frame is saved.

We have also used other Algorithms like SSIM (Structural Similarity), Histogram, Wavelet Transform but MSE (Mean Square Error) is the most precise and feasible. So, MSE is used in the project to compare frames.

```
def mse(imageA, imageB):
err=np.sum((imageA.astype("float")
- imageB.astype("float"))**2)
err /= float(imageA.shape[0]*imageA.shape[1])
return err
```

MSE Function in Python

```
z = 1
k = 1 n = 2
nf = 29168
for i in range(0, nf-1):
#while True:
a=cv2.imread("frames\\"+str(k)+".jpg",0)
b=cv2.imread("frames\\"+str(n)+".jpg",0)
m = compare_images(a,b) if m >= 5 :
cv2.imwrite("D:\\final_output_mse_5\\"+
str(z) +".jpg" , a) z = z+1 print(k,n,m)
```

k = n

n = k+1

else :

print(k,n,m) n = n+1

Module 3: Combining the non-similar frames into video sequences

Video Writer() class in python is used to capture a video, process it frame-by -frame and constructors and methods to save that video. To read frame cv2.imread() is used. Output Video is in greyscale as all the computations are done using greyscale as pixel to noise ratio is lowest in greyscale. And features of images are clearly visible and easy to distinguish in greyscale. After the optimized video is generated from resulted frames the input video and the processing data that is frames of both videos (input and optimized) are deleted.

XII. EXPERIMENTAL RESULTS

This work is implemented using Python 2.7.11. The objective of this work is to optimize the storage space occupied by CCTV footage based on redundancy of adjacent frames.

In this work, the MP4 file has been set as input video. The proposed work is mostly effective in less density areas. A video having maximum frame rate will give the better compression ratio.

The minimum hardware required is CPU to be clocked at 1.8 – 2.3 GHz Intel® CoreTM i3 processor with 4GB DDR3 RAM with HDD having more than 5 GB free for processing 500MB of video. These 5 GB will be cleared after the video is optimized.

We performed this work with input as test video of duration 20 minutes, size of 110 MB, having resolution 640 X 480 and 25 frames per second./

After compression techniques are applied, the output video obtained is of duration 7 minutes, size of 76 MB, having resolution 640 X 480 and 25 frames per second.

Due to compression, the duration of video is reduced to 65% and the size is reduced to 30.91% of memory.

XIII CONCLUSION AND FUTURE WORKS

It will design in such a way that the CCTV surveillance system can fulfil the needs of the user for particular surveillance area. It is designed for optimizing the storage for Closed-Circuit Television (CCTV) because the storage is a real challenge with increasing market demand.

This algorithm gives satisfactory results by reducing average space of any video footage by nearly 31% shown in fig 4.

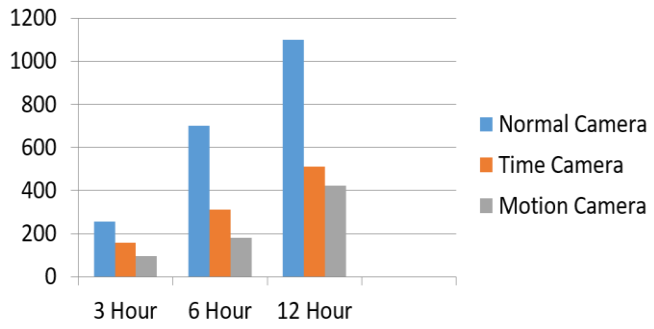

Fig 4. Comparison of memory storage of various camera

• Our method works great with low dense areas, a better method to go with high density can be done.

• Face recognition, edge detection techniques can be used to increase the efficiency.

• It can be used in army surveillance, space research and bank security. Overseeing locations that would be hazardous to human. For example, highly radioactive industrial environment.

• Arduino and Raspberry pie can be used for the implementation of these algorithms into CCTV camera.

XIV. REFERENCES

[1] Jasmeet Kaur, ACombined DWT-DCT approach to perform Video compression base of Frame Redundancy Volume 2, Issue 9, September 2012 International Journal of Advanced Research in Computer Science and Software Engineering, pp-325-332

[2] Seagate, Video Surveillance Storage: How Much is Enough? White Paper, February 2012 http://www.seagate.com/files/staticfiles/docs/pdf/whitepaper/video-surv-storage-tp571-3-1202-us.pdf

[3] Video Surveillance and Storage Survey by IHS Technology. September 2014

https://www.emc.com/collateral/analyst-reports/ihs-video-surveillance-storage-intersection-it-physical-security.pdf

[4] Xiaosong Wang, Xinyuan Huang, Hui Fu A Color-texture Segmentation Method to Extract Tree Image in Complex Scene, International Conference on Machine Vision and Human-machine Interface, 2010, pp 621-625.

[5] T.Chiang and Y.-Q. Zhang, A new rate control scheme using quadratic rate distortion model, IEEE Trans. Circuits Syst. Video Technology, Feb.1997.