NEW HORIZON
COLLEGE OF ENGINEERING
New Horizon Knowledge Park, Ring Road, Marathalli
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

Visvesvaraya Technological University
Belagavi

A Mini Project Report on

MOTION DETECTION AND STEGANOGRAPHY

Submitted by

Ananyaa Sundar – 1NH19EC006

Harikrishnan S – 1NH19EC040

Jayanth S – 1NH19EC045

Vaishnavi P – 1NH19EC139

Guided by

Ms. Ishani Mishra

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

NEW HORIZON
COLLEGE OF ENGINEERING
New Horizon Knowledge Park, Ring Road, Marathalli
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

BENGALURU-560103

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

Certified that the Mini project entitled **"Motion detection and Steganography"** is carried out by **Ms. Ananyaa Sundar** bearing **USN:1NH19EC006**, **Mr. Harikrishnan S** bearing **USN: 1NH19EC040**, **Mr. Jayanth S** bearing **USN: 1NH19EC045** and **Ms. Vaishnavi P** bearing **USN: 1NH19EC139**, bonafide students of NHCE, Bengaluru in partial fulfillment for the award of Bachelor of Engineering in Electronics and Communication of the Visvesvaraya Technological University, Belagavi during the year 2020-21. It is certified that all corrections and suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The mini-project report has been approved as it satisfies the academic requirements in respect of the mini-project work prescribed for the said degree.

Signature of the Guide	Signature of the HoD
Ms. Ishani Mishra	Dr. Sanjeev Sharma
Senior Assistant Professor	Professor & HOD
Department of ECE	Department of ECE
NHCE, Bangalore	NHCE, Bangalore

External Viva

Name of the Examiners	Signature with Date
1. _____	1. _____
2. _____	2. _____

ABSTRACT

With the number of crimes increasing day by day, the security concerns are at a surge too. Be it the physical world or the cyber world, a person feels threatened due to the presence of unusual activities and hackers. To reduce this stress, if not completely remove it from the picture, we have come up with a project that detects motion in the physical world which is combined with a method to communicate over the network in a concealed manner so as to not trigger any sort of suspicion. The first part of the project is achieved by a Motion detection algorithm and the second part is achieved with the help of Steganography. The codes of both these parts are written down in the MATLAB environment. With the help of this project, we can achieve

- The detection of any sort of movement in a frame of observation.
- Communicating with people over the network without the message being intercepted by hackers.

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along with the completion of our project. All that we have done is only due to such supervision and assistance and we would not forget to thank them.

We respect and thank Dr.Manjunatha, for providing us an opportunity to do the project work in New Horizon College of Engineering and giving us all support and guidance, which made us complete the project duly.

We are highly indebted to Dr.Sanjeev Sharma, Professor, and Head of Department, Department of Electronics and Communication Engineering, New Horizon College of Engineering, Bengaluru for the constant encouragement and providing us with all the facilities.

We owe our deep gratitude to our project guide Ms. Ishani Mishra, who took a keen interest in our project work and guided us all along till the completion of our project work by providing all the necessary information for developing a good system.

We would also like to extend our sincere esteems to all staff in laboratory for their timely support and colleagues in developing the project and people who have willingly helped us out with the abilities.

Ananyaa Sundar - 1NH19EC006

Harikrishnan S - 1NH19EC040

Jayanth S - 1NH19EC045

Vaishnavi P – 1NH19EC139

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The security concerns in recent times have grown up immensely where it is important for all of us to be able to safeguard our property from threats present outside such as thefts, destruction of property, and many more. To match the rising levels of intimidators, the surveillance techniques also need to be improved. Thus came up the methodology of motion detection.

Along with physical threats present around us, cyber threats too are a matter of concern. Sending an important piece of information can induce a huge amount of anxiety on both sides so as to keep it out of the hands of an external interceptor. To solve this hurdle to the best possible, the method of Steganography originated by which a message (may or may not be encrypted) is hidden within an ordinary file or message which cannot be detected by anyone except the people who have the needed key to access the information.

1.1 MOTION DETECTION

The process of detecting a change in the position of an object with respect to its surroundings is called motion detection. When done by natural organisms, it's known as motion perception. There are quite a lot of methods to achieve motion detection which can be categorized into three categories: electronic (using infrared rays, lasers, etc), mechanical (using switch or triggers), devices (PIR sensor, cameras). Major applications of motion detection are to detect unauthorized entry, turning lights on/off when someone enters/leaves a space, etc. These applications can be achieved by applying simple algorithms along with hardware components in various environments such as PyCharm using python, MATLAB, and quite a few more. Currently, there is Motion detection software available in the market which requires dedicated cameras or even as simple as Webcams on our laptops for its use. Some of these are cloud-based and hence are developed based on algorithms that are updated regularly for better results. Lately advances In ML and AI have enabled Motion Detection to learn and develop its own algorithms which update itself based on training datasets.

In the MATLAB environment, motion detection can be achieved using a couple of features present in the MATLAB development kit as toolboxes. The features are:

- Computer Vision:

Computer vision is a feature present in the MATLAB environment which uses images and videos to detect, classify and track objects or events in order to understand a real-world scene. In general, it is a field of Artificial Intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos, and visual inputs and take actions or come to conclusions based on the information gathered. In order for the computer to understand and recognize the distinct types of data being fed to it, it has to be calibrated and taught with vast quantities of data. The whole process can be linked to the real-world system of parents and teachers helping out kids to get to know the world they're living in by enlightening them with facts and information about the world. Computer vision now plays a huge role in various fields, from convenience stores to testing driverless cars, the healthcare sector, agriculture, banking, industrial, and many more. Top companies such as Amazon, Google, Microsoft, and Facebook have invested millions of dollars in the research and development of computer vision.

- Blob analysis:

The method of analyzing an image that has undergone binarization process i.e setting a threshold value as a standard and converting a gray image into values of 0 and 1 to focus only on the object of inspection is called blob analysis. It is the most basic method of for analyzing the shape features of an object such as the presence, number, area, position, length, and direction of the blobs (lumps). The process of blob analysis consists of three steps – Extraction, refinement, and analysis. In the first step of extraction, one of the image thresholding techniques is applied to obtain a region corresponding to the objects being inspected. The extracted region is often flawed by the noise of various kinds thus, in the next step refinement, the extracted region is enhanced using region transformation techniques. In the final step analysis, the refined region is subjected to measurements and the final results are computed. If the region represents multiple objects, it's split into individual blobs and each of them is inspected separately. Blob analysis plays an important role in object detection and related projects.

- Foreground detection and background subtraction:

This process separates objects that display changes in their positions in a series of images from their stationary background. It plays a major role in the field of image processing. Here, the moving objects are referred to as foreground, and the parts that don't change are referred to as the background. The process starts with considering a series of images that are to be analyzed. Then, the foreground detection compares these images to find out the change in positions of objects present in fame. We then get these objects removed to obtain the parts that stay still in all these compared images by applying the background subtraction process on these images. In simple terms, the

foreground is extracted from the considered images to obtain the background. This analysis finds its applications in various projects like video surveillance, optical motion capture, human-computer interaction, traffic monitoring, real-time motion gesture recognition, and many more.

1.2 STEGANOGRAPHY

The technique of hiding data within an ordinary, non-secret file so as to convey it to someone without arousing suspicion, where the secret data is then is known as steganography. In simpler terms, it's a method used to hide data in a cover file and send it to the intended person without others getting to know about the message's presence in the cover file even on going through it. Steganography can be combined with encryption as an extra step in protecting data. It is not similar to cryptography as in the latter method, the message is modified in such a fashion that the reader wouldn't understand what is actually written but in steganography, the message is completely hidden which can't be noticed by anyone seeing the cover file. The roots of steganography dates back to ancient times, 440BC where a Greek ruler by the name of Histiaeus had his most trusted servant's head shaved, marked a message on his head, and sent him to his bondsman Aristagoras once the hair regrew on the servant's head. There are five major types of steganography depending upon the nature of the cover object – image, text, video, audio, and network.

- Image Steganography:

Hiding data by taking an image as the cover object is known as image steganography. Images are widely used in digital steganography as there are a huge number of bits present in the digital form of an image. Approaches used in image steganography are Least Significant Bit Insertion, Redundant pattern encoding, masking, and filtering, encrypt and scatter, coding and cosine transformation, etc.

- Text Steganography:

Hiding information inside text files is known as text steganography. It involves changing the format of existing text, generating random character sequences, changing words within a text, or using context-free grammars to generate readable texts. Techniques used in text steganography are random and statistical generation, Format based method, and linguistic method.

- Audio Steganography:

Embedding a secret message into an audio signal which alters the binary sequence of the audio file is known as audio steganography. It is comparatively a difficult process when compared to image steganography. Various techniques of audio steganography are least significant bit encoding, parity encoding, spread spectrum, and phase coding.

- Video Steganography:

Hiding various kinds of data into a digital video is known as video steganography. The main advantage of this type is that a large amount of data can be hidden and as it is a moving stream of images and sounds, it can be considered to be a combination of image and audio steganography. Two main classes of video steganography are embedding data in an uncompressed raw video and compressing it later and secondly embedding data directly into a compressed data stream.

- Network Steganography:

The technique of embedding data within network control protocols used in data transmission such as TCP, ICMP, UDP, etc is known as network steganography.

CHAPTER 2

LITERATURE REVIEW

Title of the paper	Author & Year of Publication	Outcome	Limitation
Steganography hiding text and image in a colour image using MATLAB	G Indira Devi ; V N Sireesha February 2018	Hiding messages in an image	
Detection and tracking system of moving objects based on MATLAB	Habib Mohammed Hussien October 2014	Motion detection	

CHAPTER 3

EXISTING SYSTEM

At the present moment, the two parts of the project – Motion detection and steganography exist individually.

Motion detection: Currently motion detection finds its uses in various fields, especially in IoT based projects such as home automation where motion sensors and few other sensors are used to detect motion in a room. If there is motion detected, the lights in that room are turned on. Once there is no motion detected at the same time, no human presence detected in that room, the lights are automatically turned off. Another application of motion detection in the IoT field is security. Motion sensors are installed in buildings to detect unauthorized movement in restricted areas, particularly after hours. This can be connected to intruder alarms which can inform authorities about the movement.

Steganography: In today's world, steganography is rarely used alone. It is mostly combined with cryptography in communication sectors. However, there are quite a few areas where steganography plays an important role. Most of the modern printers print out a series of barely visible yellow dots on every page, which when interpreted properly will contain details about the print such as date and time, the printer model, and serial numbers. Just as good as the whole steganography sounds, it is also being used in the wrong ways, by actors creating malware and cyber-espionage tools as it helps them to conceal not just the data but also the fact that the data is being downloaded and uploaded. Also, it helps them to bypass the DPI (Deep Packet Inspection) systems. In order to save the world from such malware attacks, the roots of steganalysis were instilled. Steganalysis basically acts as an algorithm that detects the presence of steganography in files, whether or not a secret message has been concealed in a file.

PROBLEM STATEMENT

- To detect motion in a taken sample video.
- To note down the details of the motion such as duration of movement, the time it started and ended, frame numbers into a text file.
- To conceal the text file into a random cover image using steganography and send this image to the receiver (second person)
- On the receiver's side, to decrypt the message from the received steg image

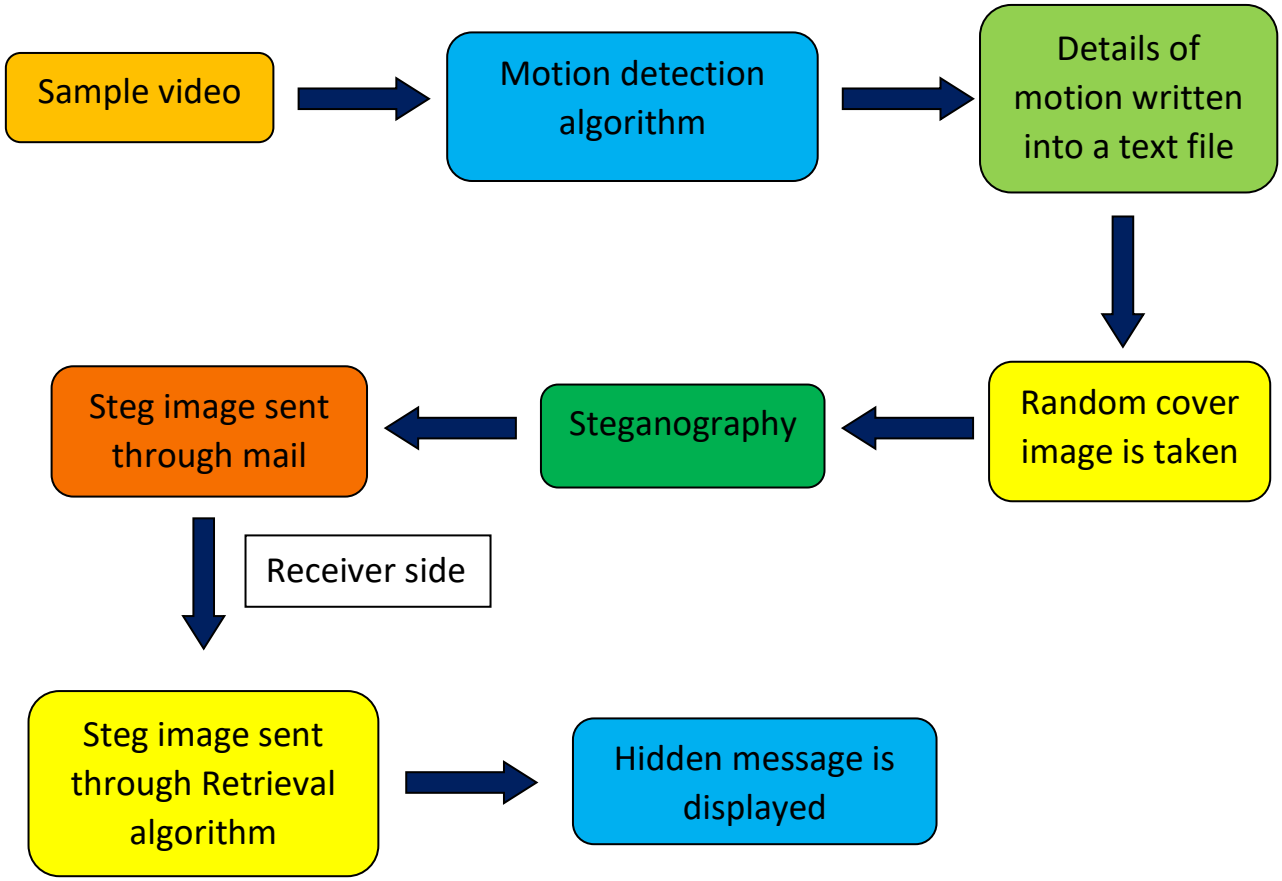
CHAPTER 4

PROPOSED SYSTEM

The project we have in view is a motion detection system which is combined with steganography.

Motion detection is achieved using features of MATLAB such as foreground detection and background subtraction and blob analysis which combined is a part of computer vision. The steganography part is achieved by normal MATLAB code.

The cover image into which the file is hidden is shared with the receiver through the mail. The receiver can then apply the retrieval algorithm to the received image to retrieve the hidden message.



SOFTWARE SPECIFICATIONS

The software used to achieve this project is MATLAB.

MATLAB, an abbreviation for Matrix Laboratory is a programming and numeric computing platform that is used by millions of engineers and scientists to analyse data, develop algorithms and create models. It finds its use in design, test, and implementation of control systems, deep learning, image processing and computer vision, machine learning, robotics, signal processing, wireless communication among its complete range of usability. Though primarily intended for numeric computing, an optional toolbox allows access to symbolic computing abilities. In addition to this, another package called Simulink can be used to add graphical multi-domain simulation and model-based design for dynamic and embedded systems.

The MATLAB system consists of five main parts:

- MATLAB language:

  It is a high-level matrix/array language with functions, control flow statements, input/output statements, data structures, and object-oriented programming features. It allows both “programming in the small” to create quick and throw-away programs and “programming in the large” to create large and complex application programs.

- MATLAB working environment:

  It is the set to tools and facilities that the user works with. It includes facilities for managing the variables in the workspace and importing and exporting data. It includes tools for managing, developing, debugging, and profiling M-files.

- Handle graphics:

  It is the MATLAB graphics system. It includes high-level commands for 2D and 3D data visualization, animation, image processing, and presentation graphics. It includes low-level commands too that allow the user to fully customize the appearance of the graphics and to build a complete Graphical User Interface (GUI).

- MATLAB mathematical function library:

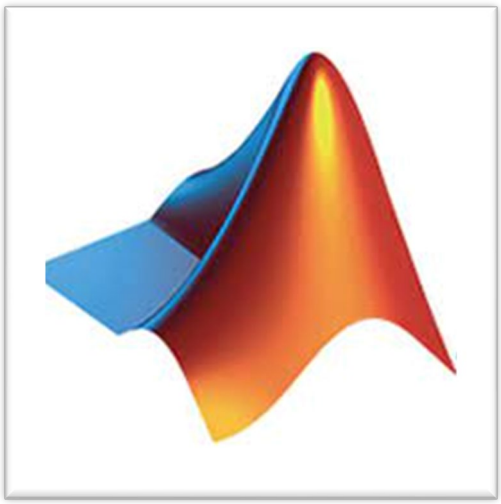
It is a vast collection of computational algorithms that ranges from basic functions such as sum, sine, cosine, and complex arithmetical commands to more sophisticated functions such as matrix inverse, eigen values of a matrix, Bessel functions, and fast Fourier.

- MATLAB application program interface (API):

It is a library that allows the user to write codes in C and Fortran that will interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MATLAB files.

Uses of MATLAB include:

- Math and computation.
- Modeling, simulation, and prototyping.
- Algorithm development.
- Scientific and engineering graphics.
- Data analysis, exploration, and visualization.
- Application development including Graphical User Interface (GUI) building.



MATLAB

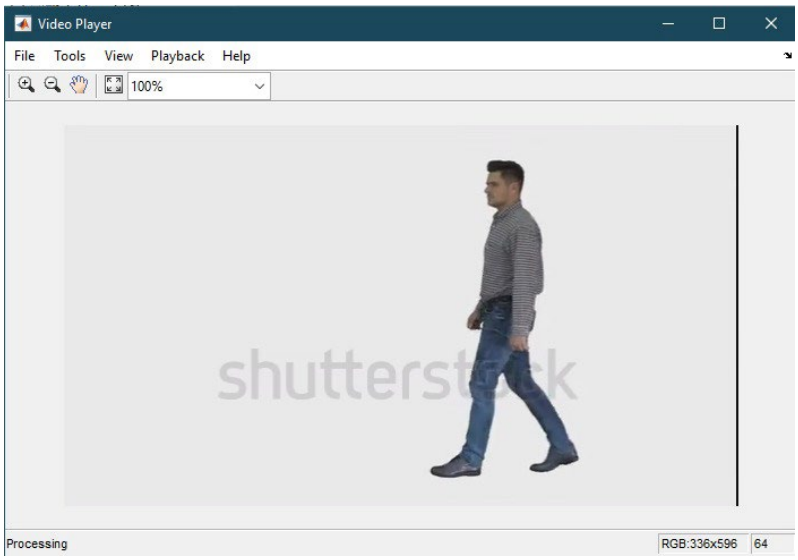
CHAPTER 5

WORKING

The project is divided into two parts – transmitter and receiver.

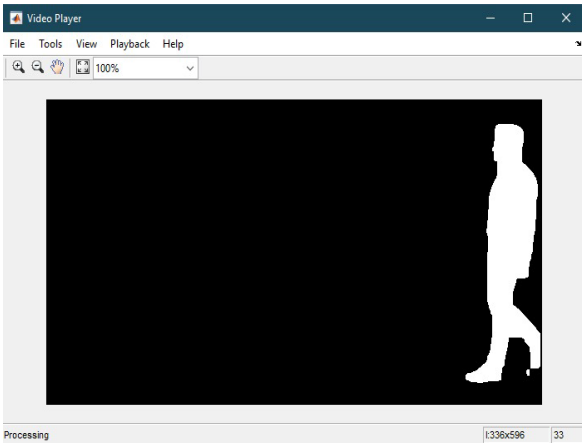
The transmitter part consists of motion detection and steganography. Assuming that both the person sending and the person receiving have the sample video with them, we just convey the details of the motion-captured to the receiver. The first step at the transmitting side is to input the sample video. After getting in the sample video, we create an object for the blob analysis, set up a video player to view the input and output videos, create a text file into which we can write down the needed details. We then initialize the mail control along with the needed variables. Foreground detection is setup including the code needed for steganography. Another video gets created which will contain only the parts with movement. The other frames which remain still are blacked out. Once the motion is detected, the details get written into the text file. Once the whole video is analysed, the text file gets saved with the updated details. The whole file then gets encrypted into the random cover image (in this case, a birthday greeting) which is then attached to the mail and sent to the ID that has been mentioned in the mail control part of the code.

The receiver part consists of the decryption program, to retrieve the hidden message from the image that had been sent to the receiver through the mail. The receiver downloads the photo from his mail, uploads it into the decryption code. Once the whole code gets executed, the hidden message gets revealed which gets stored in a text file on the receiver's device.



Actual video

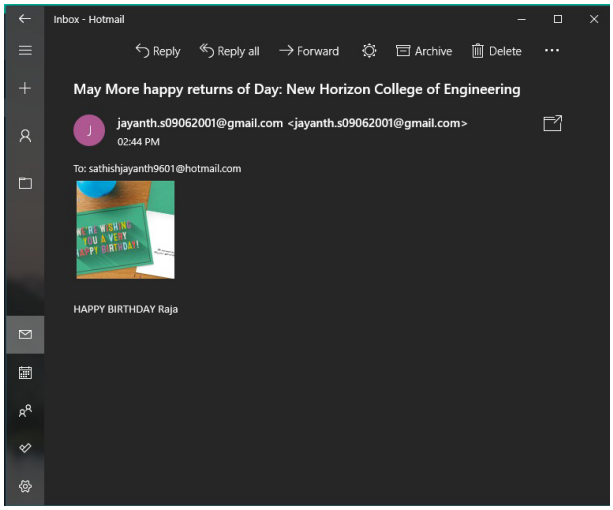
Motion indicated



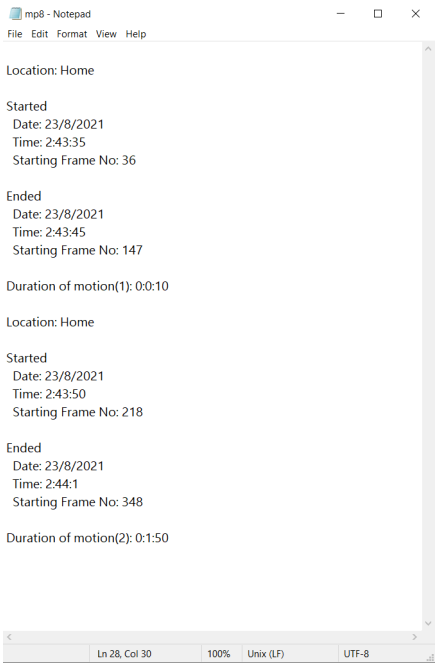
Blob analysis

Cover Image

Encrypted image



Mail with encrypted image



Details

CHAPTER 6

APPLICATION

In today's world, where security is of great concern in both the physical world and the cyber world, a solution that takes care of both at one shot has quite a lot of fields of application. At places where the whole project isn't required together, it can work as two stand-alone systems – motion detection and steganography. Few applications of the project are:

- Used by forest rangers to detect the presence of poachers and other unusual activities.
- Used by normal civilians outside their homes and official spaces to detect any unusual activities taking place outside.
- Can be used in military fields where one army can detect the motion taking place in the enemy territory and the details of this can be conveyed to the other groups situated at various other locations at the camp without arousing any suspicions.
- Used in the field of cyber security for video analysis.
- Can be used to convey information about counter-attacks.

RESULTS

Thus, a given video has been analysed for motion taking place it and the details of it are safely noted down in a text file which is then sent to the receiver by hiding it in a random cover image by the means of steganography through the mail. The receiver then gets the hidden message after applying for the decryption program on the received image. Hence, the physical world threats are surveyed with the help of motion detection and the cyber world threats of messages being intercepted are resolved by the use of steganography.

DISCUSSION AND FUTURE SCOPE

The project currently uses a sample video on which the whole inspection is done. The next step in the project will be to interface a camera which will be able to provide a live feed on which the analysis can be done. This step can enable the whole project to be used in day-to-day life. Other features such as recognition of the moving object, the speed at which it's moving, etc. The project can also be inculcated in cruise control of automated vehicles, which will control the movement of the vehicle with respect to the movement of other vehicles on the road. Under the steganography field, the whole video with the encrypted message can be sent by the use of video steganography.

CONCLUSION

The project is developed as a core code which can be applied in various fields with few modifications made according to the needs. The main aim of this project was to design a system that can monitor movements that the naked human eyes can't detect and to secretly transmit messages without arousing any kinds of suspicion, even if intercepted by hackers. Therefore, the project acts as a solution to physical and cyber world threats and can be developed further to work in a much more efficient manner and also with upgrades that will increase the boundaries the project can be used in.

CHAPTER 7

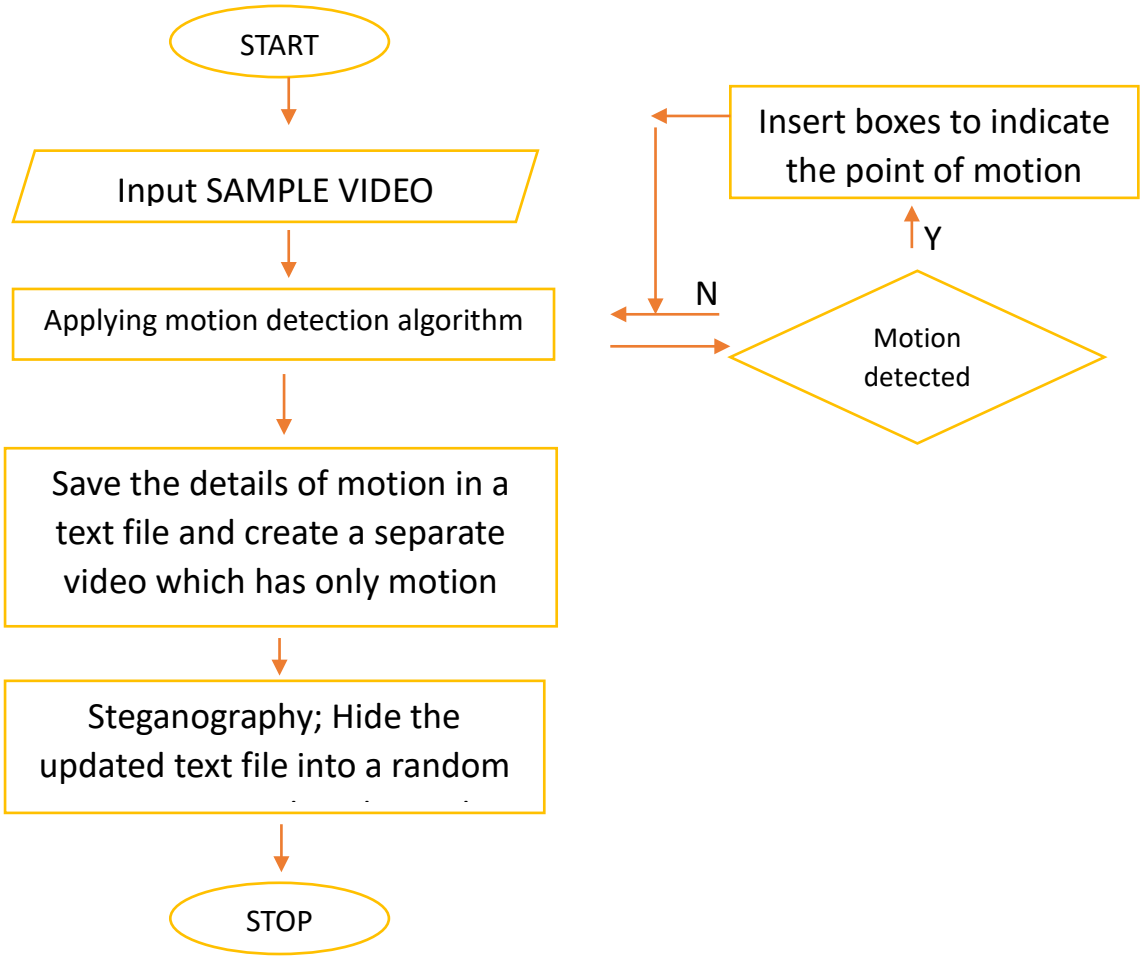
REFERENCE

- https://securelist.com/steganography-in-contemporary-cyberattacks/79276/
- https://cimss.ssec.wisc.edu/wxwise/class/aos340/spr00/whatismatlab.htm
- https://www.edureka.co/blog/steganography-tutorial#SteganographicTechniques
- https://en.wikipedia.org/wiki/Foreground_detection
- http://what-when-how.com/introduction-to-video-and-image-processing/blob-analysis-introduction-to-video-and-image-processing-part-1/
- https://en.wikipedia.org/wiki/Computer_vision

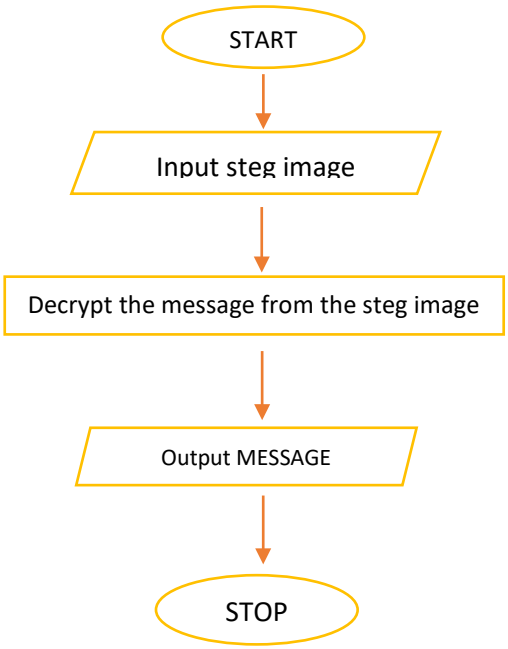
APPENDIX

FLOW CHART:

TRANSMITTER



RECEIVER



CODE:

Motion detection and steganography:

```
%% Create foreground detector object
detector = vision.ForegroundDetector("NumTrainingFrames", 20,...
                    "InitialVariance", 60*60,...
                    "MinimumBackgroundRatio", 0.7,...
                    "NumGaussians", 3);

%% Read in video file
filename = "miniv.webm";
videoObject = VideoReader(filename);
reader = vision.VideoFileReader(filename,...
                    "VideoOutputDataType", "uint8");

%% Create object for blob analysis
blob = vision.BlobAnalysis("MinimumBlobArea", 100);

%% Set up video player
player = vision.VideoPlayer("Position", [180, 100, 700, 400]);
player2 = vision.VideoPlayer("Position", [300, 200, 700, 400]);
```

```
%% video writing
mywriter = VideoWriter("mymovie.avi");
open(mywriter);

%% creating a document in txt formate named 'details.txt' to store the information collected
in the clips
fileid = fopen("details.txt","w");

%% Mail ID
mailID = "--MailId--"; %write mail id to whome u want to send mail in place of --MailId--
mailSender();

%% initialize the variables
NoFramObjDet=0;
NoFrameObjNotDet=0;
MotionState = 0; %0 means motion not taking place, 1 means motion taking place
frameNo = 0;
motionDet = false; %false means motion has stoped, thus stop writing video
                   %True means motion is taking place, thus keep writing video
NthMotionDet = 0;
StopWriting = false; %false stop writing video %True keep writing video
hr = 0;
min = 0;
sec = 0;

%% Foreground detection
%create a loop to run through video
while ~isDone(reader)

   frameNo = frameNo +1;     %counting frames
   frame = step(reader); %load next frame

   %create foreground mask
   mask = detector.step(frame);
   mask = imopen(mask, strel("rectangle", [3,3]));
   mask = imclose(mask, strel("rectangle", [15, 15]));
   mask = imfill(mask, "holes");

   %find bounding box
   [~,~,bbox] = blob.step(mask);
   %to indicate if object is detected or not
   ObjDetCount = size(bbox,1);

   %checking if there is any motion
   if(ObjDetCount>0)
```

```
      NoFramObjDet = NoFramObjDet+1; %used to count the no of frames the motion is
detected to consider it as moving object
      if(NoFramObjDet >= 20) %True of object is detected in more than 5 frames
        if((ObjDetCount>0)&&(MotionState==0))

          %intimating the detection to the owner
          if (NthMotionDet == 0)
            messageString = strcat("Hello Raja," + newline + "Wish You Many More Happy
Returns Of the Day Raja." + newline +...
                          "Hope you are doing well see you soon."+ newline + "Take care" +
newline + "Thankyou" + newline + "HR");
            %change --Happy-birthday.jpg-- image to the image that
            %you want to send
            sendmail(mailID,"May More happy returns of Day RAJA: From
HR",messageString,{'Happy-birthday.jpg'})
          end

          NoFramObjDet = 0; %setting no of frames object detected to 0
          MotionState=1; %to indicate that motion has begin (if 0 motion was not there till
then/ if 1 motion was there till then)
          motionDet = true; %used to write video when it is true
          StartingTime = clock; %reads the starting time of motion in matrix form
          if(StartingTime(4)>12) % if hours in PM set it to IST
            StartingHr = StartingTime(4)-12;
          else
            StartingHr = StartingTime(4);
          end
          %location of where motion is detected
          SLocation = "Location: Home" + newline;
          %Starting time in straing
          SStartTime = newline + "Started" + newline + "  Date: " + int2str(StartingTime(3)) +
"/" + int2str(StartingTime(2)) + "/" + int2str(StartingTime(1)) + newline + ....
              "    Time:  " + int2str(StartingHr) + ":" + int2str(StartingTime(5)) + ":" +
int2str(StartingTime(6)) + newline;
          %Frame no of motion detected.
          SStartFrameNo = "  Starting Frame No: " + int2str(frameNo) + newline;
          %The frame pic in which the motion was detected first
          figure(1)
          imshow(frame)
        end
      end
    end

    %checking if there is no motion
    if(ObjDetCount==0)
      NoFrameObjNotDet = NoFrameObjNotDet+1; %used to count the no of frames the
motion is not detected to consider that there is no motion
```

```
        if(NoFrameObjNotDet >= 5)
          if((ObjDetCount == 0)&&(MotionState==1))
            NoFrameObjNotDet=0;
            MotionState=0; %to indicate that motion has begin (if 0 motion was not there till
then/ if 1 motion was there till then)
            motionDet = false; %reads the ending time of motion
            EndTime = clock;
            if(EndTime(4)>12)
              ehr = StartingTime(4)-12;
            else
               ehr = StartingTime(4);
            end
            SEndTime = newline + "Ended" + newline + "  Date: " + int2str(EndTime(3)) + "/" +
int2str(EndTime(2)) + "/" + int2str(EndTime(1)) + newline + ....
                "  Time: " + int2str(ehr) + ":" + int2str(EndTime(5)) + ":" + int2str(EndTime(6)) +
newline;
            SEndFrameNo = "  Starting Frame No: " + int2str(frameNo) + newline;
            %%The frame pic in which the motion stopped detected
            figure(2)
            imshow(frame)
            StopWriting = true;
          end
        end
      end

      %writing the video where the motion is detected
      if(motionDet)
        %insert bounding box in frame
        ObjFrame = insertShape(frame,"rectangle",bbox,"color","r");
        %writing video
        writeVideo(mywriter,im2double(ObjFrame));
      end
      %inserting empty frames to indecate the ending of the video
      if(~motionDet && StopWriting)
        for i=1:30
           writeVideo(mywriter, zeros(videoObject.Height,videoObject.Width));
        end
      end

      % duration of motion
      if(~motionDet && StopWriting)
        %documentation of detections
        NthMotionDet = NthMotionDet+1;
        hr = EndTime(4)-StartingTime(4);
        min = EndTime(5)-StartingTime(5);
        sec = EndTime(6)-StartingTime(6);
```

```
        SMotionDuration = newline + "Duration of motion(" + int2str(NthMotionDet) + "): " +
int2str(hr) + ":" + int2str(min) + ":" + int2str(sec) + newline;

        final = strcat(SLocation + SStartTime + SStartFrameNo + SEndTime + SEndFrameNo +
SMotionDuration);

        fprintf(fileid,"%s", newline ,final);

%       sending mail with details
%       sendmail(mailID,"Details of the detected Motion",final)

        hr = 0;
        min = 0;
        sec = 0;
        StopWriting = false;
        DetailsMail = true;
    end

    %update video player
    player.step(frame);
    player2.step(mask);
end
%% sending the details of the detection after processing in encrypted form
EncryptMess();
%change --WishVM.jpg-- image with your encrypted image
sendmail(mailID,"May More happy returns of Day: New Horizon College of
Engineering",'HAPPY BIRTHDAY Raja',{'WishVM.jpg'});

%% %% Clean up
fclose(fileid);
close(mywriter);
release(detector);
release(reader);
release(blob);
release(player);
release(player2);

%% mail control
function mailSender()

mailID = "--MailId--"; %write mail id from which you want to send mail, in place of --MailId--
setpref("Internet","SMTP_Username",mailID);
setpref("Internet","SMTP_Password","--WriteYouMailIdPassword--"); %write the mail id
account password in place of --WriteYouMailIdPassword--
setpref("Internet","E_mail",mailID);
setpref("Internet","SMTP_Server","smtp.gmail.com");
props = java.lang.System.getProperties;
```

```
props.setProperty("mail.smtp.auth","true");
props.setProperty( "mail.smtp.starttls.enable", "true" );
props.setProperty("mail.smtp.socketFactory.class",...
                "javax.net.ssl.SSLSocketFactory");
props.setProperty( "mail.smtp.socketFactory.port", "465" );

end
%% encrypt
function EncryptMess()

candidate_Image=imread('--imageWhichWillHoldMessage--'); %add the name of the image in
which you want to encript the information
DocWithMess = 'details.txt';
secret=fopen(DocWithMess,'rb');        %To open the secret file
[DocWithMess,L]=fread(secret,'ubit1');       %To read secret file as bin array
% L is the length of the secret file
[n,m]=size(candidate_Image); % n= width, m=height*3
m=m/3;
%m*n is the max size to save the data
if (m*n*3<L)
   msg=msgbox('your picture is too small', 'size error', 'error', 'modal');
   pause (1);
   if (ishandle(msg))
      close(msg);
   end
end
latest_data=candidate_Image;
count=1;

for i=1:m  % width
   for j=1:n  % height
      for k=1:3  %RGB
         latest_data (i, j, k)=candidate_Image (i, j, k)-mod (candidate_Image (i,j,k), 2)
+DocWithMess (count, 1);
         if count==L
            break;
         end
         count=count+1;
      end
      if count==L
         break;
      end
   end
   if (L==count)
      break;
   end
end
```

```
%change --WishVM.jpg-- to the name by which you want to save your encripted image
imwrite (latest_data, 'WishVM.jpg', 'bmp');

CC=DocWithMess;
count1=1;
for i=1:m
   for j=1:n
      for k=1:3
         CC (count1) =latest_data (i, j, k)-candidate_Image (i, j, k);
         if count1==L
            break
         end
         count1= count1+1;
      end
      if count1==L
         break
      end
   end
   if count1==L
      break
   end
end

end
```

Decryption code:

```
%change the name --WishVM (1).jpg-- to the name of your encripted image
hid_pic=imread ('WishVM (1).jpg');
[n1, m1]=size (hid_pic) ;
ml=m1/3;
ct=1; temp=M;
%To get the secret information
for i=1:m1
   for j=1:n1
      for k=1:3
         temp (ct, 1) =temp (ct, 1) +mod (hid_pic (i, j, k), 2);
         if temp (ct, 1) ==2
            temp (ct,1)=1;
         end
         if ct==L
            break;
         end
         ct=ct+1;
      end
      if ct==L
```

```
            break;
        end
    end
    if ct==L
        break;
    end
end
c=0;  j = 0;
fileID=fopen('output.txt', 'w');
for i=1:L
    c=c+temp (i, 1)* (2^j);
    j=j+1;
    if j==8
        j=0;
        fwrite(fileID,c,'char');
        c=0;
    end
end
fclose(fileID);
```