SUMMER INTERNSHIP REPORT

Gaze maps for video sequences:

use of eye tracker to record the gaze of viewers of video sequences

Submitted by

Raju Shrestha

CIMET

July-August, 2009

Gjøvik University College, Norway

Objective

The main objective of this internship work is to track the gaze of viewers on video sequences and construct gaze maps from the eye tracker recorded data. This allows us to study how eye tracking equipment can be used to track how people watch videos and get idea on areas of interest.

Eye tracking and Gaze map

Eye tracking is the process of measuring either the point of gaze (“where we are looking”) or the motion of an eye relative to the head. An eye tracker is a device for measuring eye positions and eye movement. Eye trackers are used in research on the visual system, in psychology, in cognitive linguistics and in product design. There are a number of methods for measuring eye movement.

Eye trackers measure rotations of the eye in one of several ways. Among several types of eye trackers, non-contact, optical method is used here for measuring eye motion. In this method, light, typically infrared, is reflected from the eye and sensed by a video camera or some other specially designed optical sensor. The information is then analyzed to extract eye rotation from changes in reflections. Video based eye trackers typically use the corneal reflection (the first Purkinje image) and the center of the pupil as features to track over time. A more sensitive type of eye tracker, the dual-Purkinje eye tracker, uses reflections from the front of the cornea (first Purkinje image) and the back of the lens (fourth Purkinje image) as features to track.

Eye movement is typically divided into fixations and saccades, when the eye gaze pauses in a certain position, and when it moves to another position, respectively. The resulting series of fixations and saccades is called a scanpath. Most information from the eye is made available during a fixation, but not during a saccade. The central one or two degrees of the visual angle (the fovea) provide the bulk of visual information; the input from larger eccentricities (the periphery) is less informative. Hence, the locations of fixations along a scanpath show what information loci on the stimulus were processed during an eye tracking session. On average, fixations last for around 200 ms during the reading of linguistic text, and 350 ms during the viewing of a scene. Preparing a saccade towards a new goal takes around 200 ms.

Eye trackers necessarily measure the rotation of the eye with respect to the measuring system. If the measuring system is head mounted, as with EOG, then eye-in-head angles are measured. If the measuring system is table mounted, then gaze angles are measured.

Watching a user in real-time is interesting, but the speed of movement makes it hard to keep track of what users see and what they miss. A series of erratic eye movements suggest that a user was confused by a disorganized stimulus. While a series of controlled eye movements show that a user was looking at interesting locations. The density of these movements helps us to establish their level of concentration and comprehension.

Under natural viewing conditions humans tend to fixate on specific parts of the image that interests them naturally. The term gaze, in general, is used to mean focused attention. Mapping such gaze positions over the original stimuli (image/video) is known as Gaze maps. Gaze maps are usually represented as heat maps, as they are perhaps the most revealing of all the outputs from an eye tracking study.

Heat maps use a graded colour scheme to show visual activity. Warmer colors reveal areas that most users looked at, while colder colors show areas that few users noticed. Black reveals areas that no one looked at.

Video Gaze map

A number of studies have been made with eye tracking on images and their gaze maps. Video gaze map is relatively new and has endless uses in real world businesses. If you are interested in taking your market research business or psychological practice to the next level, then you need video eye tracking. Using eye tracking, you can process more accurate responses and information from the human brain and use the data to analyze a patient or product.

Video eye tracking is an exciting technology. It is relatively new and has endless uses. Application for video eye tracking includes medical and market research, experimental brain science examination, mental testing and treatment and much, much more. We can learn almost endless information about the brain through eye movements.

Video eye tracking is already popular in many fields. A lot of doctors and psychiatrists use the technology to better understand their patient’s needs and learn more about their behaviors. It is also being used in market research by many companies to better serve customers by learning more specifically what individuals want out of different products. It can be very useful in lie detection as well as for security purposes. The technology might even go as far as to allow you to get psychiatric analysis through your internet connection or participate in market research that is in-depth and convenient for everyone.

In this internship work, gaze mapping of sequence of videos are constructed from eye tracker's data on video sequences. The analysis of gaze map can help make a deeper understanding of the subjective salient perception which in turn helps in exploring the use of video eye tracking in the above mentioned applications.

Experimental setup

The experimental setup comprises three categories: hardware, software, test videos and test subject sitting arrangement.

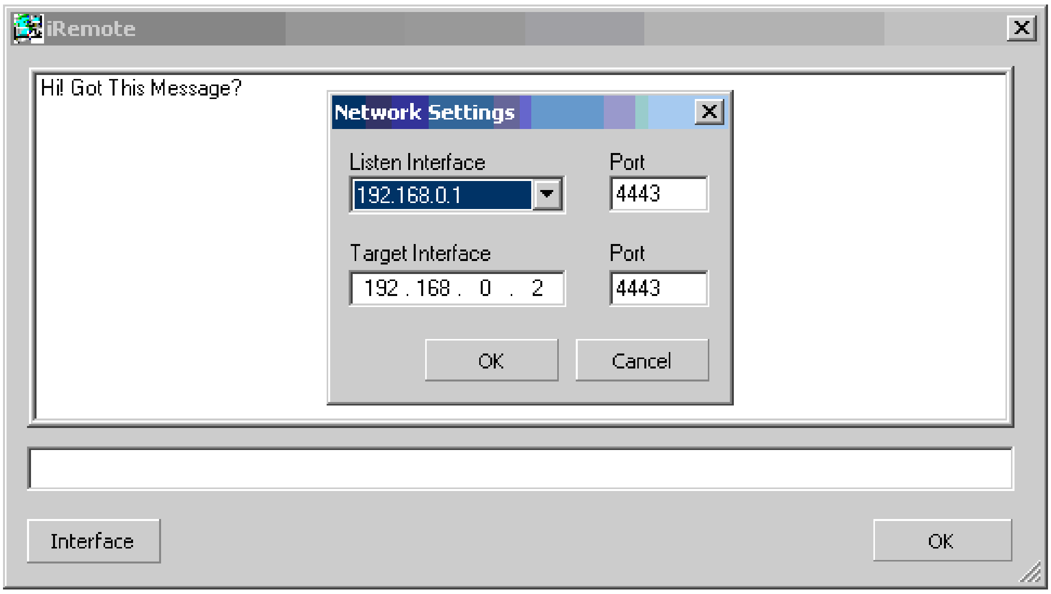
Hardware setup:

The hardware setup consists of following components:

1. The iView X RED pt camera system: SensoMotoric Instruments (SMI) RED pt camera has been used. The following figure shows the camera.

2. Stimulus PC and monitor for stimulus presentation: Test stimulus is presented on the monitor of the stimulus PC.
3. iView X workstation PC and monitor: The test subject’s eye movement is recorded in this PC by iView X software while the test subject views the test stimulus on the stimulus PC monitor.

The RED pt camera is connected to the iView X Workstation PC. The stimulus PC and and workstation PC are connected via local network or crosspatch cable. Communications between two computers can be tested using iRemote tool that comes with the SMI software by sending text message from one computer to another and see if the message appears in another computer’s iRemote console. IP address and port numbers are set accordingly in both computers as shown in the figure.



Hi! Got This Message?

Listen Interface: 192.168.0.1
Port: 4443

Target Interface: 192 . 168 . 0 . 2
Port: 4443

If the iRemote consoles receive the message typed in another computer, it indicates the correct network connection.

Software setup:

A custom software program module (**RSRunExp.m**) is developed in Matlab for the experiment that presents sequence of videos as stimuli to the subject one-after-another and controls the iView X workstation for recording the eye tracking information automatically.

The iView X software is run in the workstation computer and configured properly for calibration and recording the eye tracker data.

iRemote tool is used for testing the network connection between two computers and WinCAL tool is run in the Stimulus PC before starting the calibration. Both of these tools come with the SMI iView X software bundle.

Test videos:

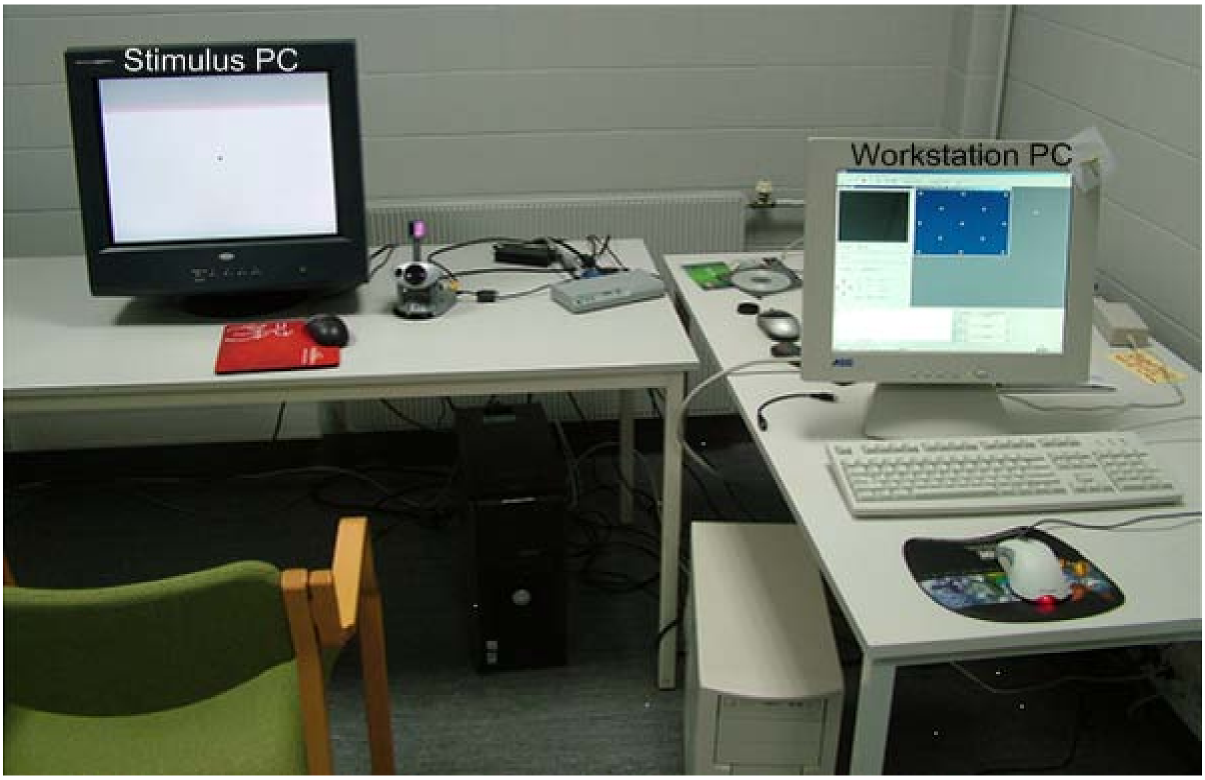
13 Test videos are obtained from University of Jean Monnet, Saint Etienne, France and 3 test videos are created here in Gjovik University College, Norway. Thus, total of 16 test videos are used. List of test videos used in the experiments are given in Appendix B.

Computer and Test subject sitting arrangement:

The area around the stimulus computer screen should be relatively free of distractions. Typically eye tracking setup places the stimulus and iView X monitors in a position where both are visible to the researcher but only the stimulus monitor is visible to the test person.

Test person is placed in a comfortable position in front of and centered to the Stimulus PC monitor. A chair should be arranged which helps minimize the movement of upper body parts of the test person. This will decrease the possibility that the test person will change their position in a way that causes gaze inaccuracies. In particular, the test person should be prevented from changing the distance from the eye to the screen during a test. It would be better to have a chair which allows the test person to rest the chin that prevents head and body movement. However due to unavailability of such a chair, a simple chair without wheels and pivots is used. The positions of the chair legs are well marked so that the distance between the monitor and the head remains within the required range.

The following figure shows real arrangement used in the experiment.



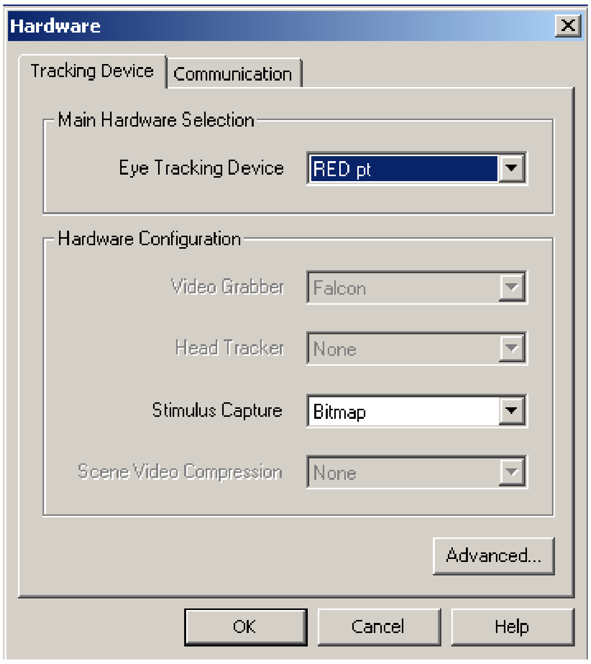
Setup Configurations:

The following setup configurations are made and used in the experiment.

- IP of the iView X Workstation is set to 192.168.0.1
- IP of the Stimulus PC is set to 192.168.0.2

Other settings are made in iView X workstation software:

- From **Setup>Hardware** menu,
  - Set Tracking device is set to RED pt:



Hardware

Main Hardware Selection

Eye Tracking Device: RED pt

Hardware Configuration

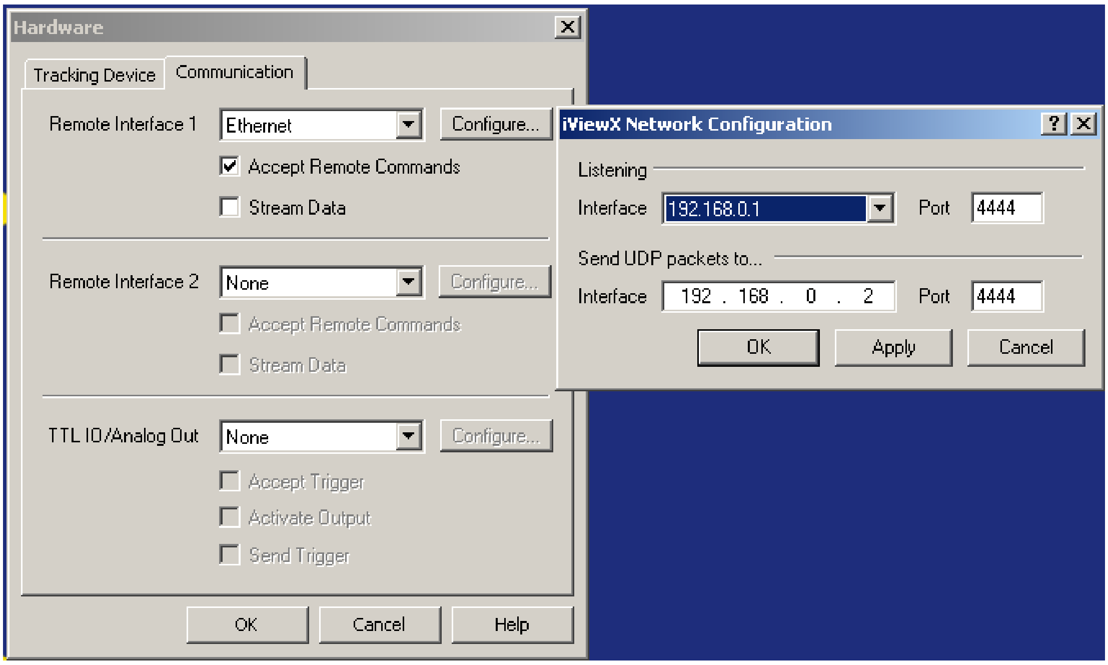
Video Grabber: Falcon

Head Tracker: None

Stimulus Capture: Bitmap

Scene Video Compression: None

  - Setup IP to 192.168.0.1 and Port number to 4444 (***must be the same as the one given for WinCal in stimulus PC***).



Hardware

Remote Interface 1: Ethernet

☑ Accept Remote Commands

☐ Stream Data

Remote Interface 2: None

☐ Accept Remote Commands

☐ Stream Data

TTL IO/Analog Out: None

☐ Accept Trigger

☐ Activate Output

☐ Send Trigger

iViewX Network Configuration

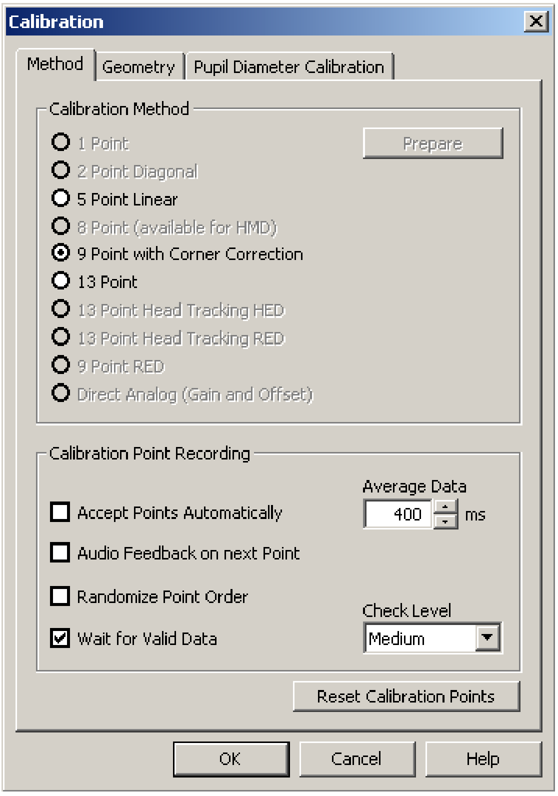
Listening

Interface: 192.168.0.1 Port: 4444

Send UDP packets to...

Interface: 192 . 168 . 0 . 2 Port: 4444

- From **Setup>Calibration** menu, set
  - Display resolution to the resolution of the stimulus pc monitor display (1024x768 in our case)
  - Calibration method used : 9 Point with Corner Correction
  - Check Level: Medium
  - Head to Monitor Distance: 700mm



Calibration Method

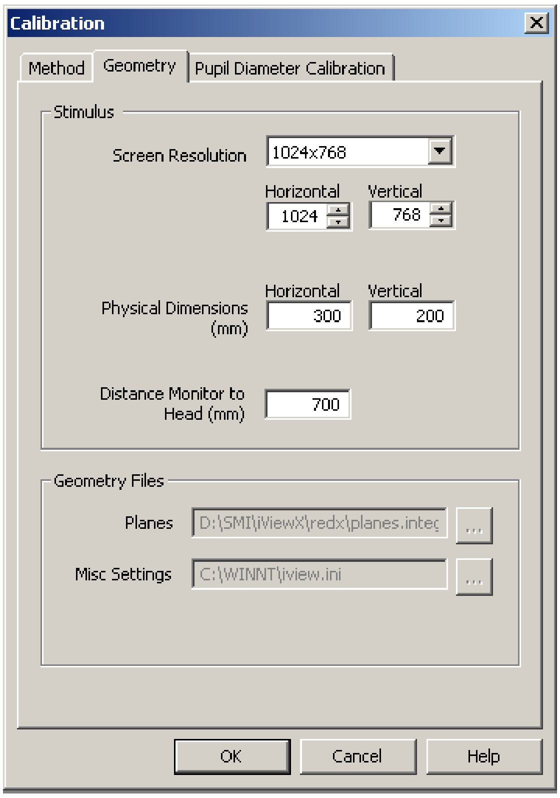
○ 1 Point
○ 2 Point Diagonal
○ 5 Point Linear
○ 8 Point (available for HMD)
● 9 Point with Corner Correction
○ 13 Point
○ 13 Point Head Tracking HED
○ 13 Point Head Tracking RED
○ 9 Point RED
○ Direct Analog (Gain and Offset)

Calibration Point Recording

□ Accept Points Automatically
□ Audio Feedback on next Point
□ Randomize Point Order
☑ Wait for Valid Data

Average Data: 400 ms

Check Level: Medium



Stimulus

Screen Resolution: 1024x768

Horizontal: 1024 Vertical: 768

Physical Dimensions (mm): Horizontal: 300 Vertical: 200

Distance Monitor to Head (mm): 700

Geometry Files

Planes: D:\SMI\iViewX\redx\planes.inte(

Misc Settings: C:\WINNT\iview.ini

- Manual Acceptance during calibration (Unmark automatic accept in Calibration menu).

Psychophysical Considerations:

Before performing the experiment, it has been discussed with psychophysicist Prof. Frode regarding experimental setup on psychophysical aspects and requirements. The following considerations were made based on his advices.

1. The amount of stray light in the area of the test person is limited to avoid confusion due to other corneal reflections. Light level changes during the test are avoided, as large pupil size changes can cause inaccuracies in the gaze data and false pupil diameter readings. The brightness level of the calibration is set to match that of the stimulus. For this, gray background has been used instead of white.

2. Camera position can be moved to best view the eye.
3. Dominant eye is used during calibration. However, this was not possible all the time when it was not possible to track that eye during the experiment due to technical difficulties like blocking by nose.

   Dominant eye of a person was determined by means of a psychophysical experiment.
4. Movement of the body and the head to be avoided during the experiment. This was the most difficult part faced because of unavailability of the appropriate chair.
5. Test videos are ordered in such a way that no two similar videos played one after another to minimize/avoid habitual errors.
6. The basic information about the test subjects: name, sex, age, eye color, wearing glasses or not, experience with eye tracking experiment and dominant eye were noted.
7. The subjective quality evaluation of the experiment with a particular subject was made by giving values from 1 to 5; with 1 indicating the best and 5 indicating the worst.

Now, with all the experimental setup and configurations readied, the experiments are carried out next.

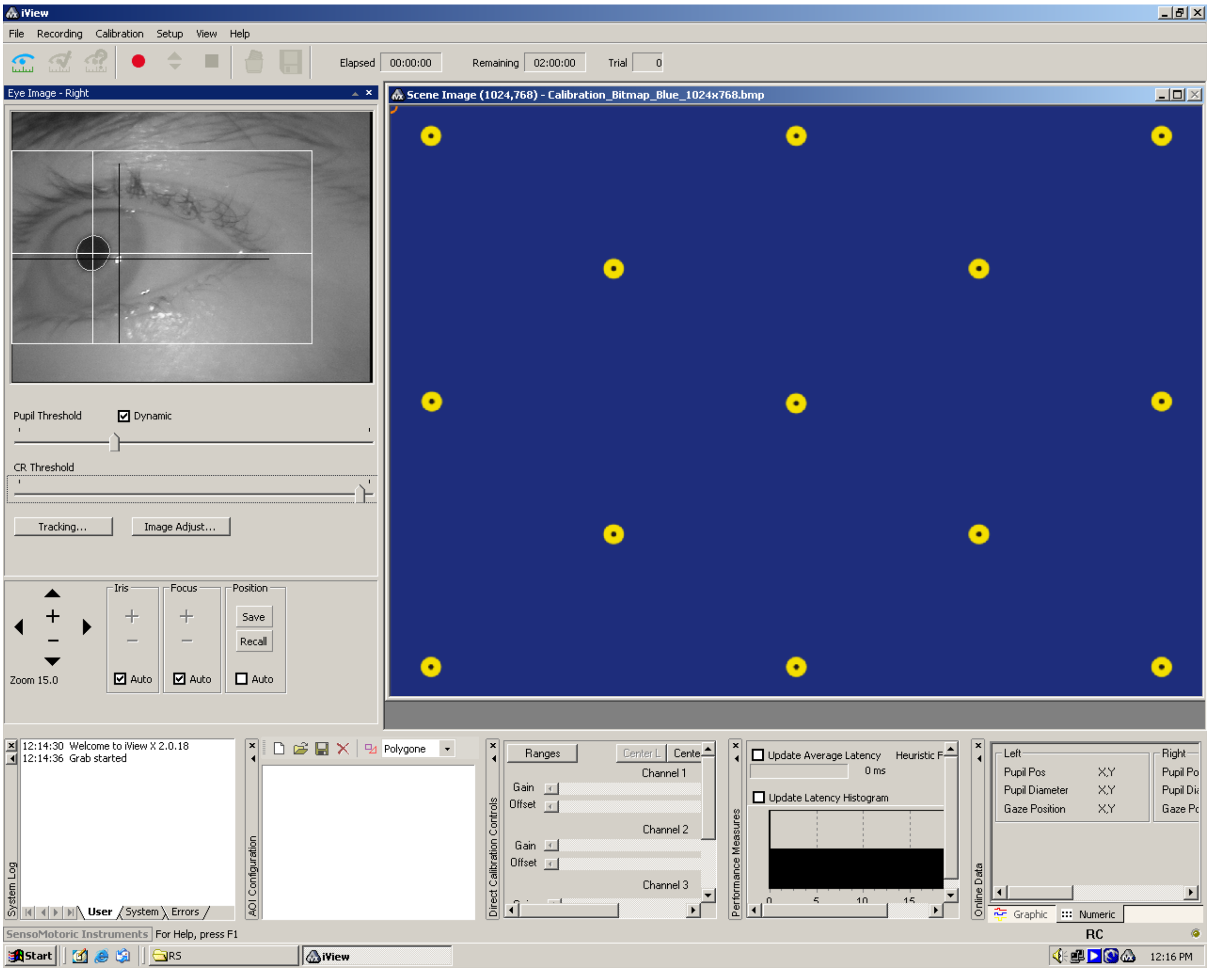
Experimental Procedure

There are 3 steps in the experimental procedure: Eye tracking, calibration & validation and Watching video and recording eye tracking data. These three steps will be carried out sequentially one-after-another. If everything goes smoothly, it took less than 10 minutes to complete with the given number of test videos. Instruction sheet (see. Annex A) for the test persons are prepared with the incorporation of the matter from the psychophysicist Prof. Frode.

The main role of the subject (in all three parts) is to look at the monitor, importantly staying still as much as possible as shown in the figure below:

1. Eye tracking

In this part, the subject will be asked to look around the stimulus monitor area as asked and the eyes will be tracked in the workstation computer. For this, the rectangular box is sized (From Tracking… button and AOI tab) so as to cover the subject's dominant eye. The Iris and Focus are set to auto by marking check boxes as shown.



Elapsed 00:00:00 Remaining 02:00:00 Trial 0

Scene Image (1024,768) - Calibration_Bitmap_Blue_1024x768.bmp

Zoom 15.0

```
12:14:30  Welcome to iView X 2.0.18
12:14:36  Grab started
```

Update Average Latency 0 ms

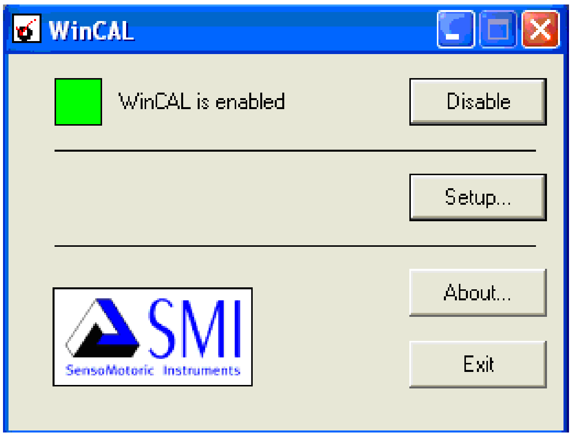
Slider for the pupil threshold is placed in such a position such that the circle covers the pupil area well and white cross mark appears with the center lies on the center of the pupil. Next, the CR (Corneal Reflection) threshold is adjusted such that the black cross mark points to the corneal reflection and it appears near the white cross mark and becomes stable. The subjects could be asked to look around the corners of the screen and see if the white and black cross marks are stable with circle around the pupil area in all positions. Then the position is saved by clicking the save button and the auto mark is checked for the position so that the camera system automatically compensate for the eye position with little movement of the head.

The RED pt camera may need to be moved up down depending upon the subject so as to make the eye tracking perfect.

2. Calibration & Validation

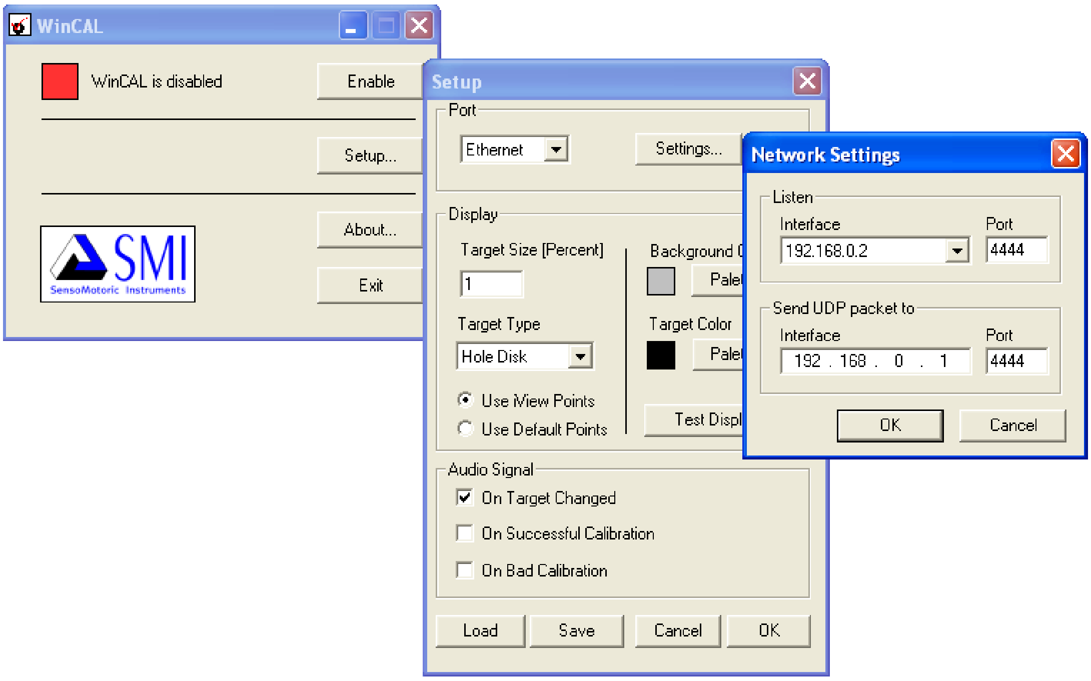
After eye tracking, the next step will be the calibration and validation. This is very important step in this experiment.

To start calibration, WinCAL is run in the stimulus PC and set in enabled mode.



🟩 WinCAL is enabled

Make sure from setup that proper IP and port number is set from the setup:



🟥 WinCAL is disabled

Port: Ethernet

Display

Target Size [Percent]: 1

Background ▨

Target Type: Hole Disk

Target Color ■

● Use View Points

○ Use Default Points

Audio Signal

☑ On Target Changed

☐ On Successful Calibration

☐ On Bad Calibration

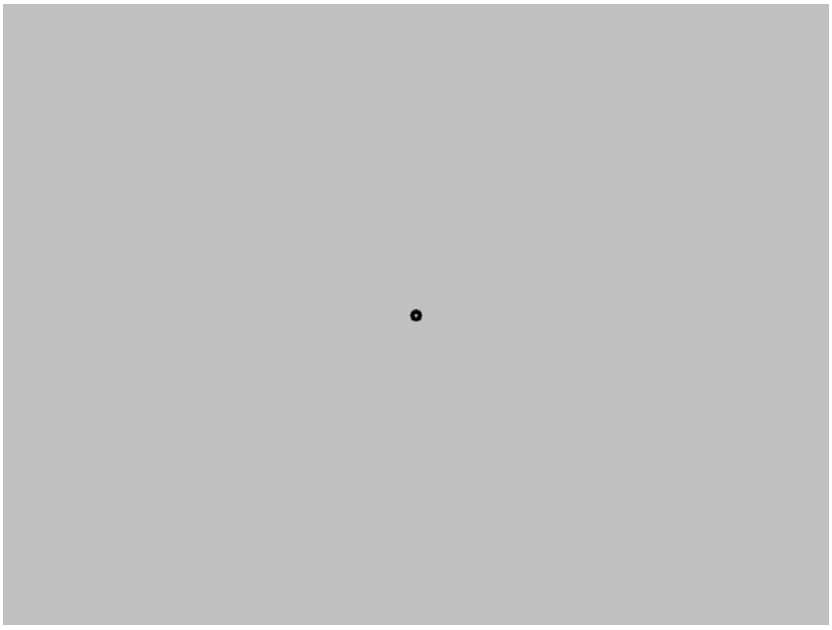
Listen

Interface: 192.168.0.2 Port: 4444

Send UDP packet to

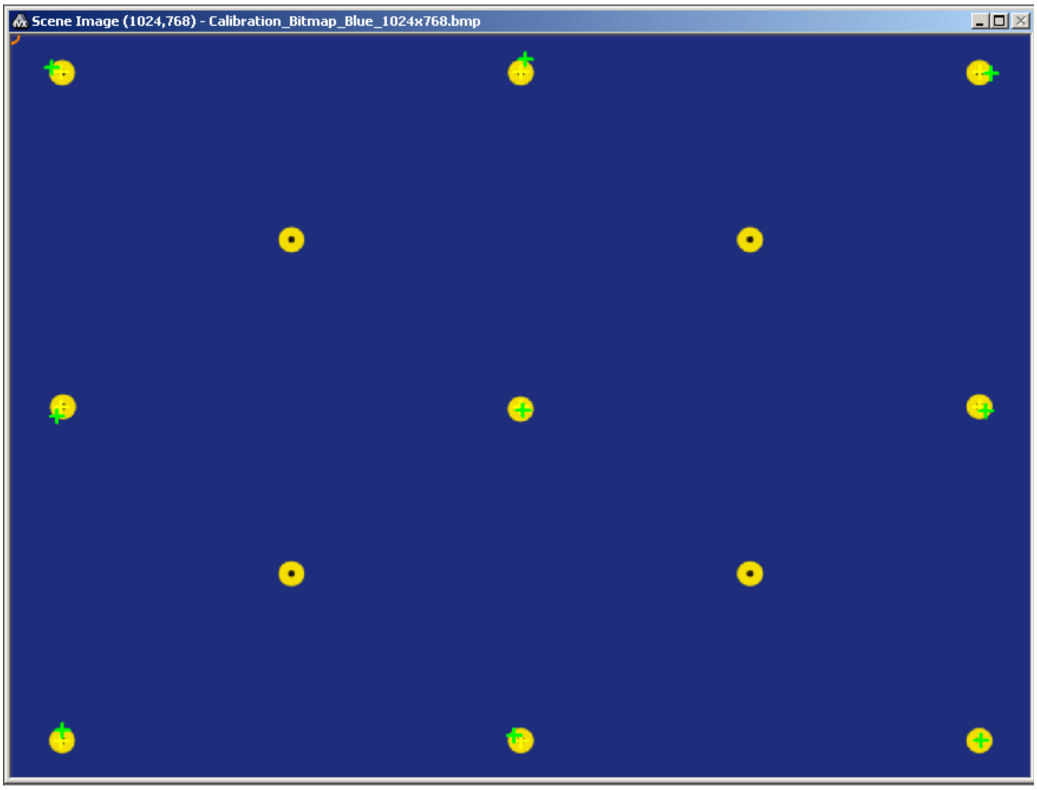
Interface: 192 . 168 . 0 . 1 Port: 4444

Then, the calibration is started by clicking **Calibration>Start** menu in the iView X software in the workstation computer. During calibration, you will see small circle on the screen as shown below. The subject is asked to look at the circle, of course by avoiding the head and body movement.



When the subject fixated on the point and fixation become stable (◈ mark indicates stable fixation, while + Indicates no fixation or lost pupil), press F6 or <space> to accept the point. The system then automatically moves to the next calibration point. Repeat this process until all 9 calibration points are accepted.

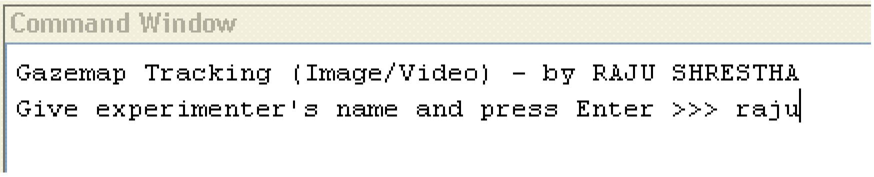
Immediately after, the same procedure will be repeated for validation by clicking **Calibration>Validation** menu, as the name indicates, to validate how well the calibration was. The green cross mark shows the validated points as shown below.


Scene Image (1024,768) - Calibration_Bitmap_Blue_1024x768.bmp

Nearer the validated points to the original points, better was the calibration. If the validated point(s) appear to be far away from the original points, it indicates poor calibration and in such case, we need to do calibration again.

3. Watching video and Record Eye tracker data

Next comes the real experiment with our test videos. For this the custom Matlab program module **RSRunExp.m** is executed. It asks for the test person's name as shown below:



```
Gazemap Tracking (Image/Video) - by RAJU SHRESTHA
Give experimenter's name and press Enter >>> raju
```

Given the name and press Enter. This starts the video on full screen mode. All 16 test videos will be played sequentially one after another. The subject needs to watch the videos as normal videos. The program controls the iView X software and instructs it to record the movement of the subject's eye while watching the videos. There will be small pause between each video (for the time taken for loading the next video) and try to be still even during these pauses. During test pauses, the recording of the eye tracking system will also be paused.

On completion, "Experiment is Completed, Thank you" message will be shown in the Matlab command window. This completes the experiment for the subject.

The recorded data will be saved in the given output path or the default path (D:\RS) in the workstation computer. The file name will be given in the following format:

`person'sname-date(yyyymmdd)Ttime(HHMMSS).idf`

An example file is `raju-20090729T121217.idf`

The experiments have been performed with 20 test subjects available at the moment during the internship work. The information about all test subjects is given in the Annex C. There are corresponding 20 eye tracking data for the test videos.

Construction of Video Gaze Map

Eye tracker data recorded during the experiment is processed and the video gaze maps are constructed from the original videos and processed data. For this, another custom Matlab program module **RSGazeMap.m** has been developed. The Annex B gives the program code and the details specification of the program module.

The eye tracker data files in .idf format are converted into .txt files by using IDF converter available with SMI iView X software. Appendix D details about the format of these .txt files.

Upon executing the **RSGazeMap.m** program module, it reads the eye tracker data for each test video for all test subjects and constructs the video gaze map frame-by frame. A delay of 400ms from the test subjects viewing of a video frame to the recording of the corresponding eye tracking data is compensated to obtain eye tracker data corresponding to a frame. The algorithms used in constructing video gaze map are given below.

Video Gazemap Construction Algorithm:

```
RSGazeMap:
For each video
      For each frame in the video
            Get frequency map for each position from all subjects
            Construct gaze map using the frequency map and Gaussian filter
            Add the gaze mapped frame to the gaze mapped video
      End For
End For
Save the gaze mapped video
```

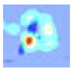
For a particular frame, the starting time and end time (in microseconds) in eye tracking data are computed by adding delay time as described above and corresponding frame time to the starting time. The following expressions show the time calculations.

```
tstart = GazeData.starttime + delay + (frameno-1)/video.rate*1000000 (µs)
tend = GazeData.starttime + delay + frame/video.rate*1000000(µs);
```

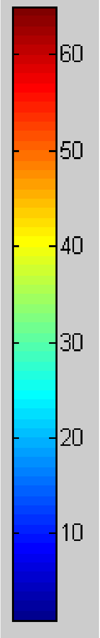
Delay value of 400ms is used in our case. The value is obtained empirically by testing the synchronization of the gaze maps with the most likely viewing region of interest in the video frames.

Gaze data within these time ranges are used in calculating frequency maps. Frequency maps are obtained using two approaches so that resulting gaze maps could be compared.

Approach 1: In this approach number of gazes at a particular pixel is summed up for all the test subjects and the gaze map is then constructed from the aggregate frequency map.

Gaze maps are normally shown with heat map representations (like ), and so did here. For this Gaussian filter is applied to the frequency map to obtain corresponding Gauss map. The hsize of [200 200] and standard deviation of 10 are used as shown in the following Matlab statements. These values are empirically obtained for better representation of heat maps.

```
gauss = fspecial('gaussian',[200 200],10);
GaussMap = imfilter(FreqMap,gauss,'replicate');
```



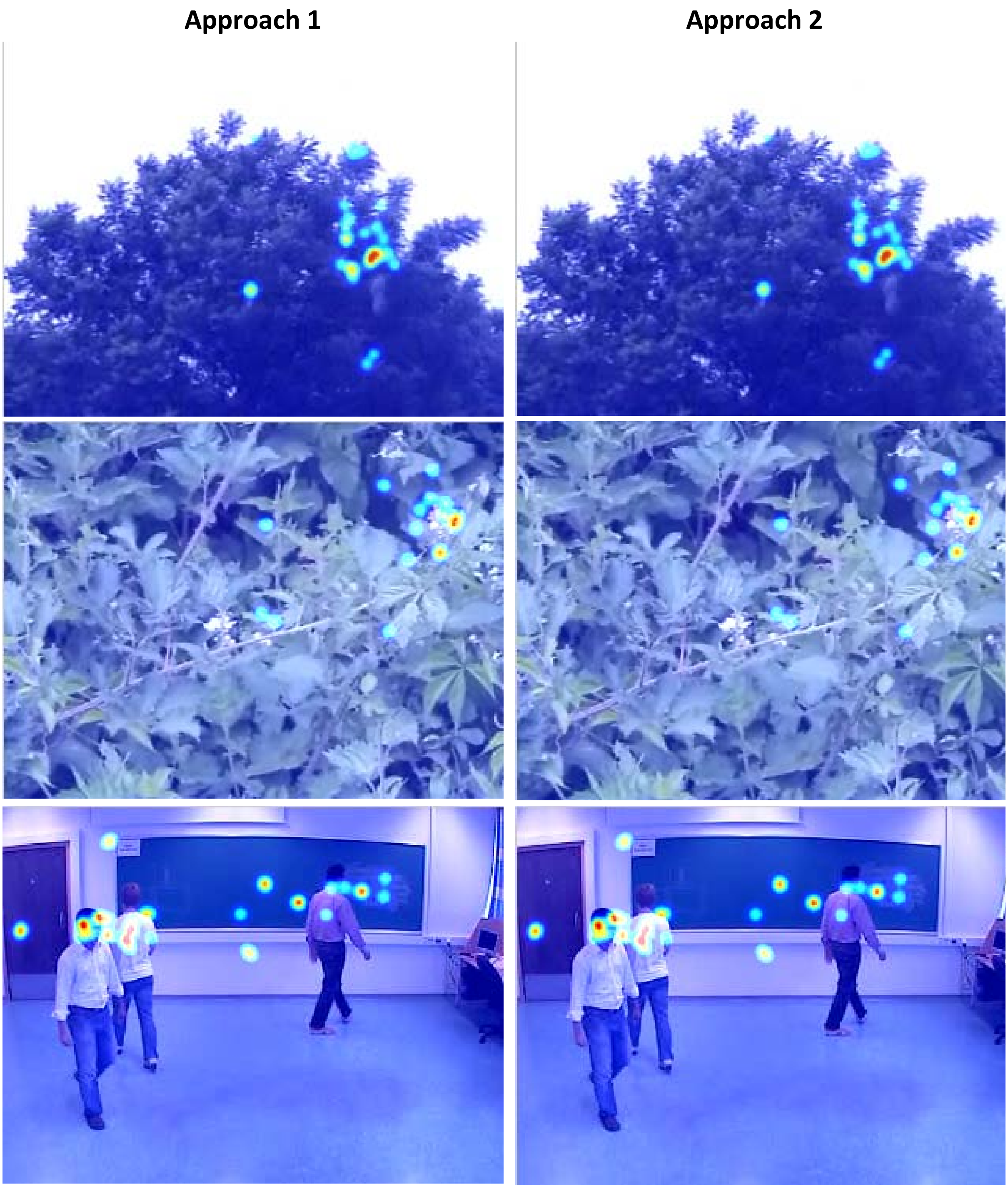
The Gauss map, thus obtained are normalized using the maximum and minimum values and the normalized Gauss map is reshaped into three-dimensional color data using JET color map in Matlab having 64 color levels (as shown in the color bar at the right side). The Matlab statements used for this purpose are given below.

```
clevels = 64;
n = min(GaussMap(:));
x = max(GaussMap(:));
if x == n, x = n+1; end
cdata = round((GaussMap-n)/(x-n)*(clevels-1)+1);
cdata = uint8(reshape(cmap(cdata(:),:),[size(cdata) 3])*255);
```

Since the original video sizes (320x240 in most of the videos) are smaller than the full screen view (1024x768) used during the experiment, the Gauss map obtained are scaled back to the original video frame size. For this reason, the hsize parameter in the Gauss filter is used to be bit higher ([200 200]). This heat map representation of gaze data is combined with the original video frame to obtain the gaze mapped video frame by taking the minimum of sum of the pixel value in the original video frame and the maximum possible value (i.e. 255).

Approach 2: Here, instead of summing up the frequency of gaze at a particular point for all the subjects, the frequency map is obtained individually for all the subjects and corresponding Gauss map is obtained from them using the similar technique of Gaussian filter as described above in the first approach. The individual Gauss maps, thus obtained, are then summed up to obtain resultant Gauss map. Next, the gaze mapped video frame is obtained from this Gauss map in the same way as in the first approach.

Following figures shows comparative (side-by-side) results of gaze map video frames with the two approaches.

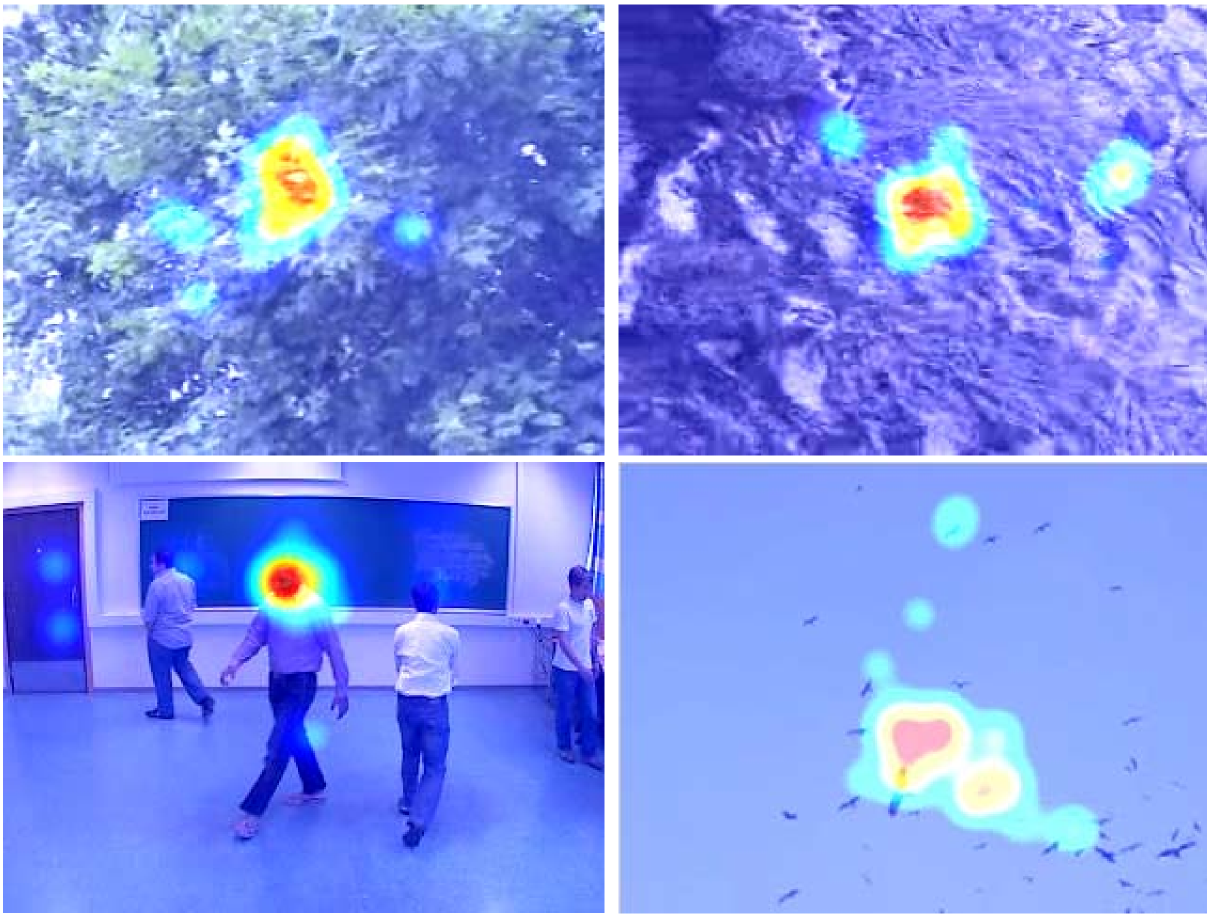


Approach 1

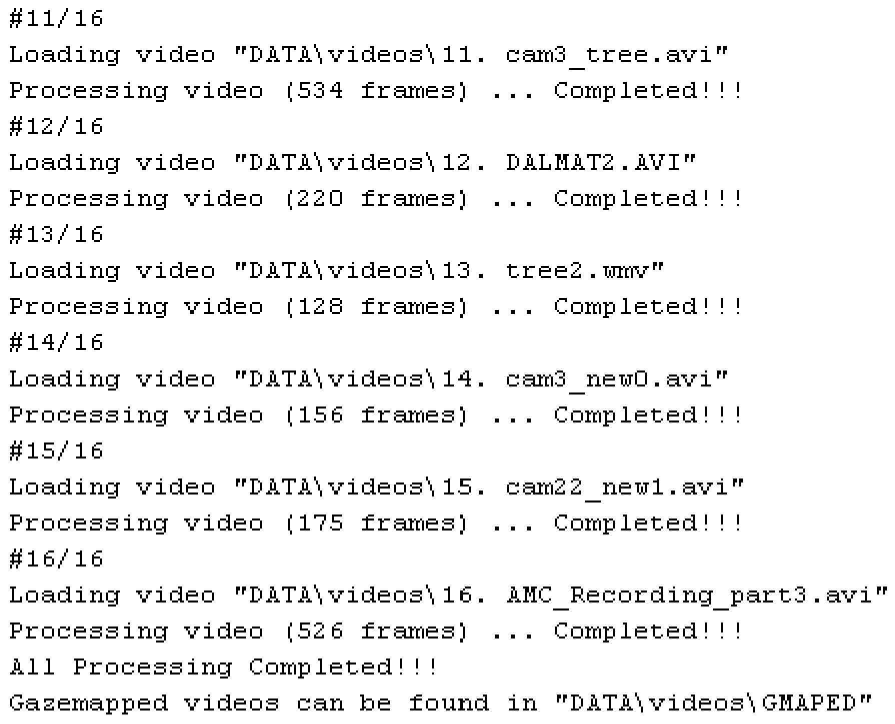
Approach 2

From these results we see that the resulting gaze mapped video frames are the same with both the approaches. However, the second approach takes quite longer time for processing compared to the first approach. So, the first approach is recommendable.

With the bigger size of hsize and sd parameters, the scattered heat maps could be made merged to form bigger heat maps showing bigger area of interests. The following figures shows some of the resulting gaze mapped video frames with hsize = [300,300] and sd = 30.



The program displays the name of the video file and the frame under processing so as to indicate the progress on gaze map construction program execution. The following screenshot shows an example display.



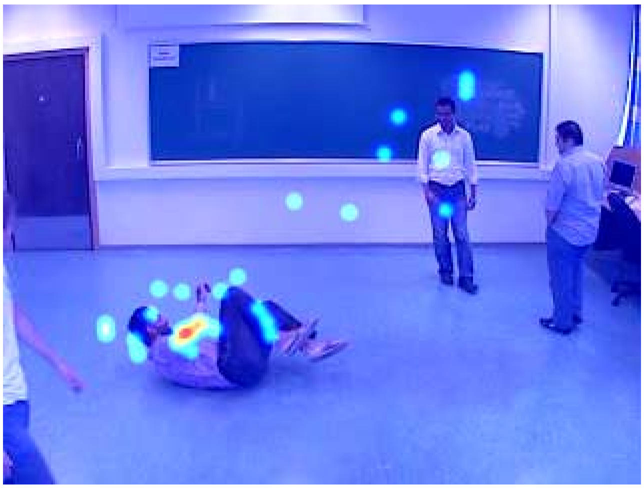
```
#11/16
Loading video "DATA\videos\11. cam3_tree.avi"
Processing video (534 frames) ... Completed!!!
#12/16
Loading video "DATA\videos\12. DALMAT2.AVI"
Processing video (220 frames) ... Completed!!!
#13/16
Loading video "DATA\videos\13. tree2.wmv"
Processing video (128 frames) ... Completed!!!
#14/16
Loading video "DATA\videos\14. cam3_new0.avi"
Processing video (156 frames) ... Completed!!!
#15/16
Loading video "DATA\videos\15. cam22_new1.avi"
Processing video (175 frames) ... Completed!!!
#16/16
Loading video "DATA\videos\16. AMC_Recording_part3.avi"
Processing video (526 frames) ... Completed!!!
All Processing Completed!!!
Gazemapped videos can be found in "DATA\videos\GMAPED"
```

The gaze mapped frames obtained are then added together to form the gaze mapped video of the original video. The gaze mapped video will be saved in the given folder or the default folder 'DATA\VIDEOS\GMAPED'. The file name will be given the same as the original video file name except with '_gmap' added at the end (like '`01. cam3_new3_`**`gmap`**`.avi`'). Appendix C describes the input parameters that can be passed into the **RSGazeMap.m** program module.

The gaze mapped video can be played to see and analyze how the people look at the videos. In order to see the gaze map more clearly frame by frame, we could play the videos slowly. For this yet another Matlab program module **RSPlayGazemapVideo.m** is provided. It allows playing the video by providing the frame rate at which you want to. Appendix C describes its usage. Appendix B also shows snapshots of a gaze mapped video frame for each of the 16 original test videos.

Discussion of the Results

From the results of the gazed video map, we see that the gaze positions may differ from subjects to subjects depending upon many factors like his/her interest, mood, situation etc. Consequently we can see different gaze positions for the same video frame. However, when there are some areas of interest (e.g. falling down of a man, flying butterfly), the subjects tend to gaze at those areas. Testing with large number of subjects we obtain heat map concentrated on those areas. This allows us to ignore non-important areas and focus only on those interesting areas in different applications. The following gaze map video frame illustrates this fact.



In this video frame, a person is falling down and this is the interesting thing for a viewer. The concentration of the heat map shows this.

We also see in this experiment that some of the things that move fast (like flying leaves, thrown stick) in the videos are not trackable. Moreover, the eye doesn't seem to move in synchronous with fastly moving objects.

With 20 test subjects, we could get some general idea about the areas of interest in the videos. With large number of test subjects, we could have even more precise results.

Difficulties and Recommendations for improving Results

As the experiments were carried out under the available resources and environment rather than the required ideal environment, the results could be improved significantly with the improvement of the arrangements. Followings are some of the difficulties faced during the experiments that might have caused experimental errors and that could be improved with better arrangements.

- As the experiment is very sensitive to the movement of the head and the body of the subjects which were unavoidable in our experiments. Some of the subjects find very difficult in avoiding such movements causing not so good calibration. This problem was faced a lot with many subjects even with multiple tries.
- The RED pt camera needs to be moved up and down; here and there so as to make the tracking of the eye better and was difficult in our case. This could be eased with some stand that can be raised and lowered.
- Also in order to track the dominant eye, the eye tracker needs to be moved at the left and right sides of the monitor, which were avoided in our experiment. Rather, the comfortable eye was tracked instead of the dominant eye. This could be corrected with proper arrangement.

With these problems solved, perhaps, we may be able to use automatic acceptance during the calibration resulting much more accurate results.

Some other difficulties faced during the experiment are:

- Blue eyes are difficult to track with the RED pt camera system.
- Running the experiment software (**RSRunExpt.m**) after calibration in the Stimulus PC distracts the test subject. It could be avoided with the arrangement that the program can be executed automatically after calibration or remotely.
- Large video files couldn't be used as Matlab couldn't handle them.

Appendix A

Instruction Sheet for Subjects

Objective: We are going to check out how eye tracking equipment can be used to track how people watch videos.

There are 3 steps in this experiment: Eye tracking, calibration & validation and Watching video. These three steps will be carried out sequentially one-after-another. If everything goes smoothly, it will take less than 10 minutes to complete. The main role of the subject (in all three parts) is to look at the monitor, importantly staying still as much as possible. However, eyes can be hovered around the monitor. Three very important points to be noted are:

- **Be comfortable and relaxed**
- **Try not move the body and the head; and**
- **Position the eyes at a distance of about 50cm from the monitor**

The instructions to be followed during three steps are given below:

1. Eye tracking (~2 mins)

In this part, the subject will be asked to look around the monitor area as asked and the eyes will be tracked in another computer. It will take around 2-3mins.

2. Calibration & Validation (~3 mins)

After eye tracking, the next step will be the calibration and validation. This is very important step in this experiment. During calibration, you will see small circle on the screen which will be moved to different positions on the monitor. You need to focus your eyes on those points, of course keeping your head and body stationary.

Immediately after, the same procedure will be repeated for validation, as the name indicates, to validate how well the calibration was.

3. Watching video (~5 mins)

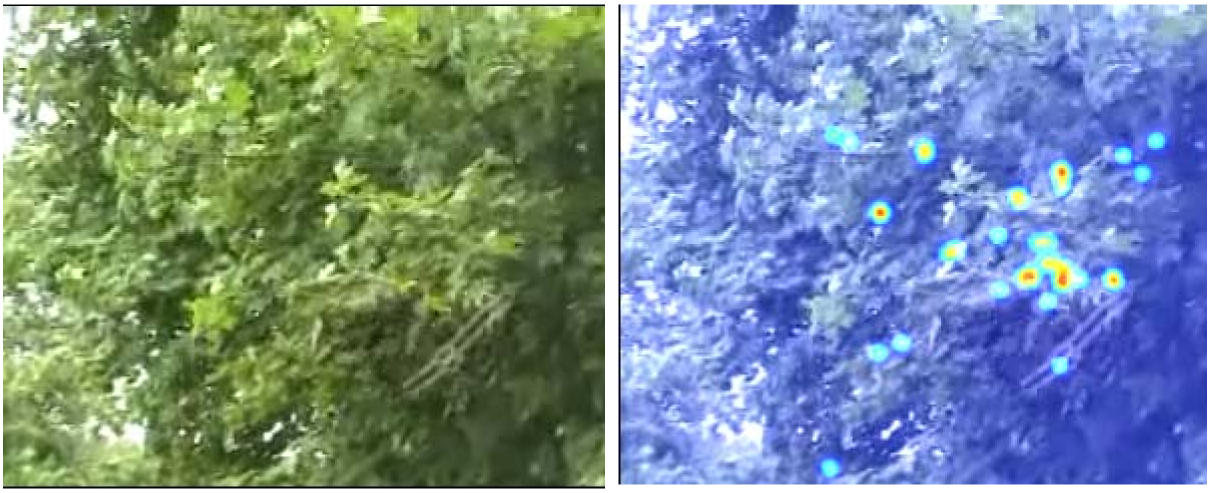
Once the calibration and validation step is done well, a sequence of videos will be played on the monitor. ***Just watch videos as you watch normal videos. There will be small pause before each videos and try to be still even during these pauses***. On completion, “Experiment is Completed, Thank you” message will be shown in the Matlab command window.

Appendix B

Test Videos and Gaze maps

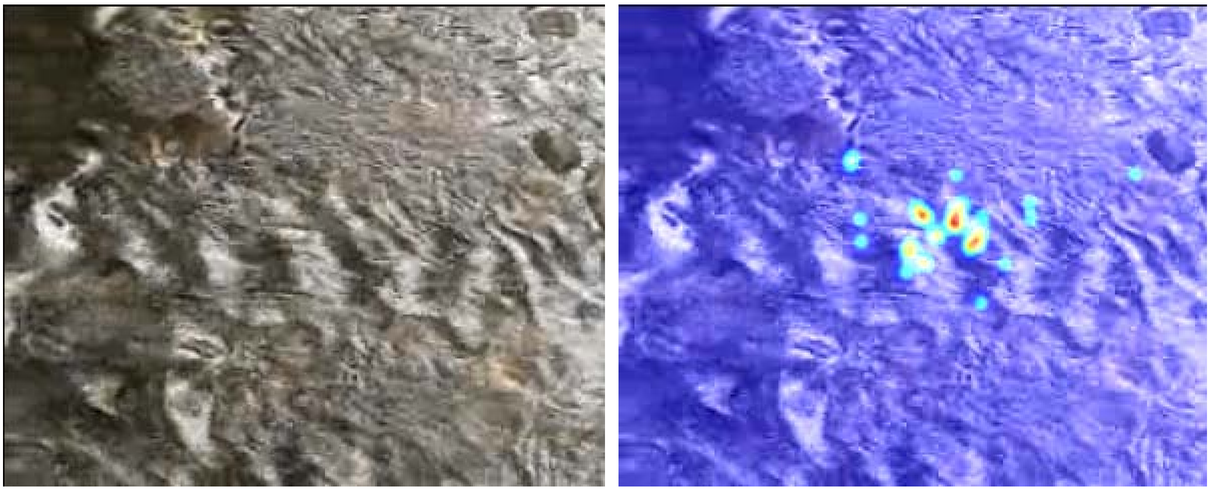
The list of test videos used and in order of their playing in the experiment along with snapshots showing salient feature and corresponding gaze mapped frames. For complete gaze map of videos, the gaze mapped videos are provided along with this report.

01. cam3_new3.avi



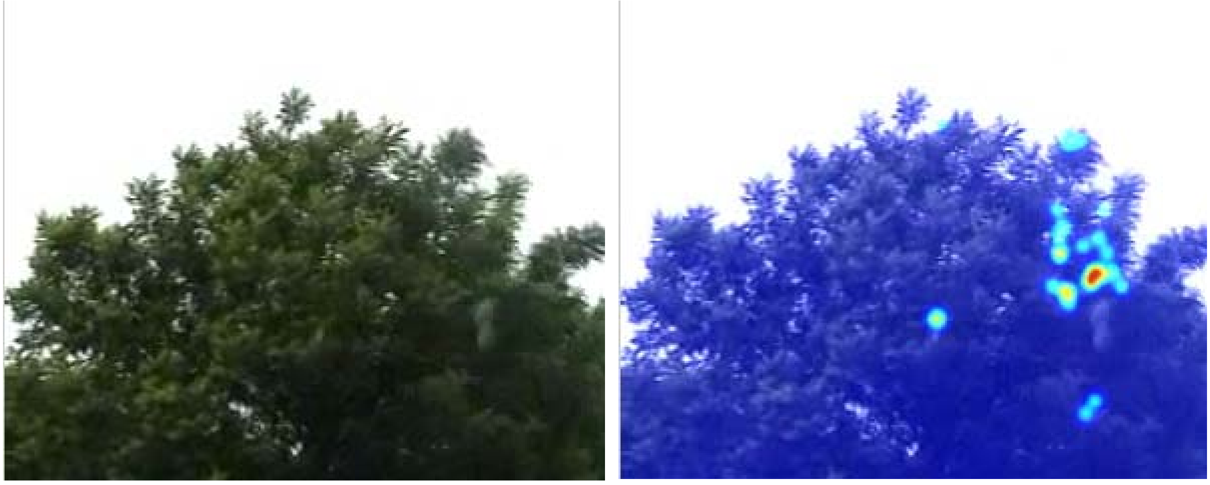
Salient Feature: Thrown stick

02. cam3_4th.avi



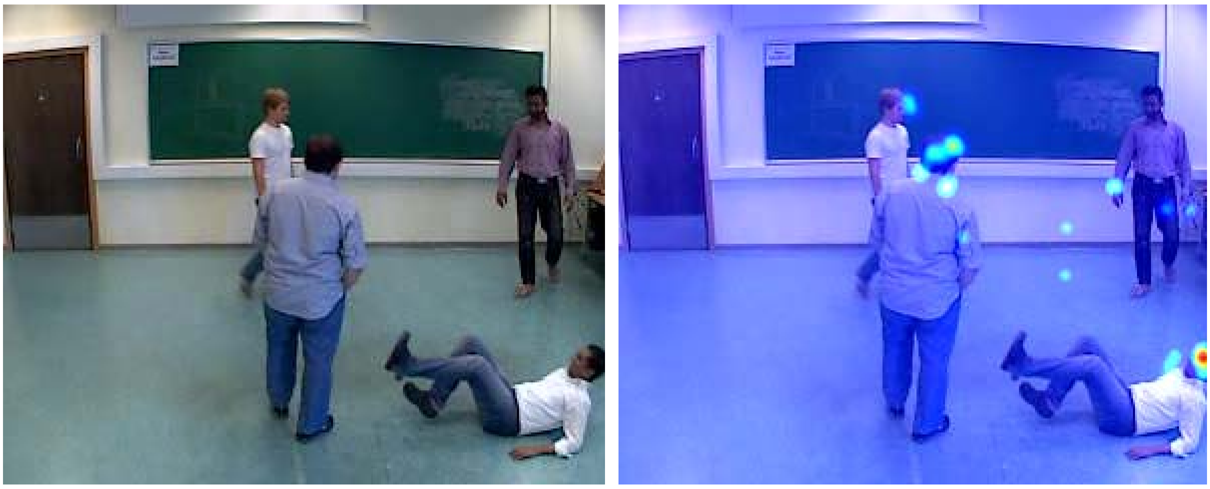
Salient Feature: Flowing water in the river

03. tree1.wmv



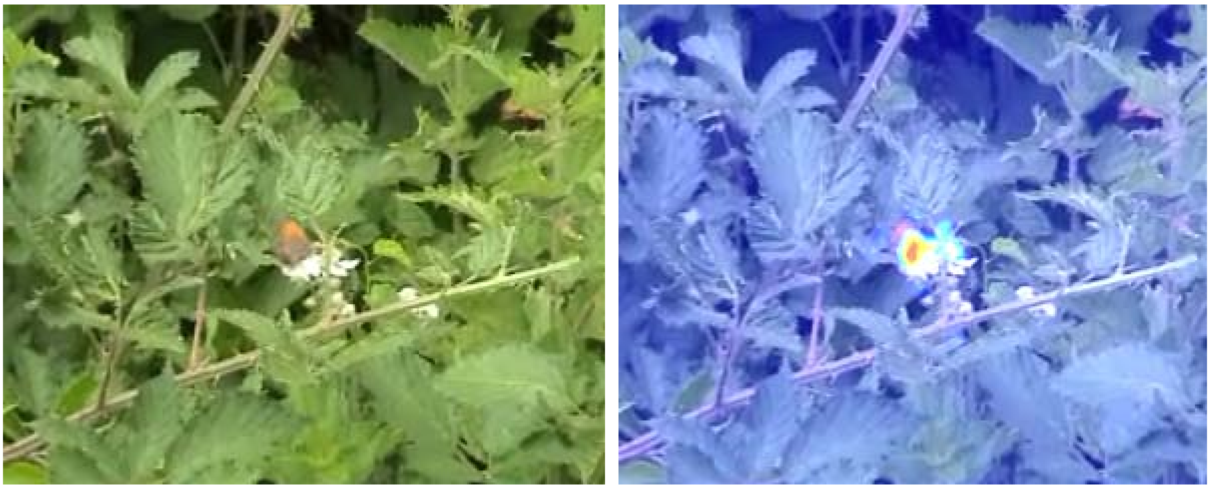
Salient Feature: Something falling down

04. AMC_Recording_part1.avi



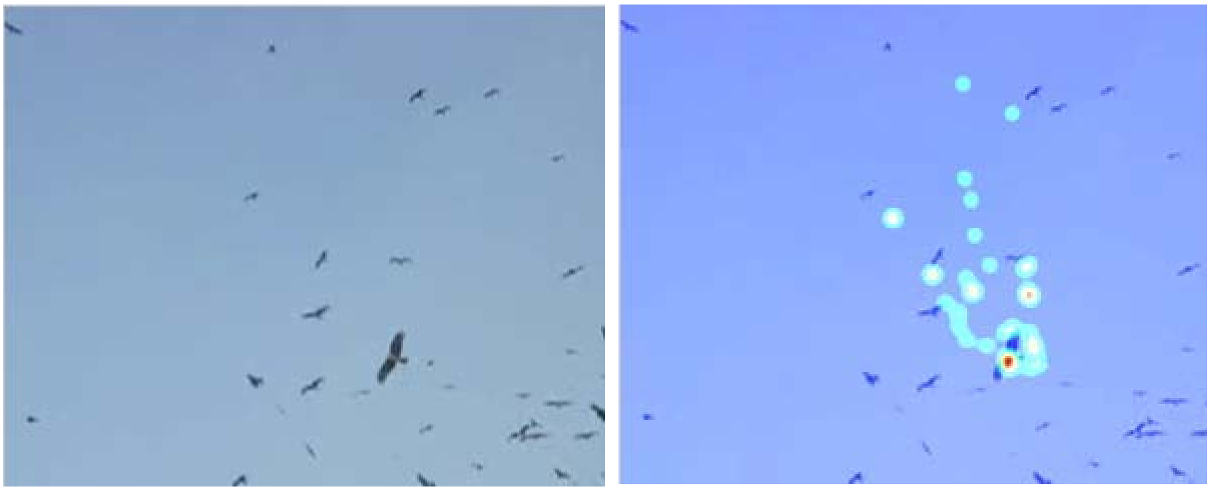
Salient Feature: A person falling down on the floor

05. cam3_butterfly_new1.avi



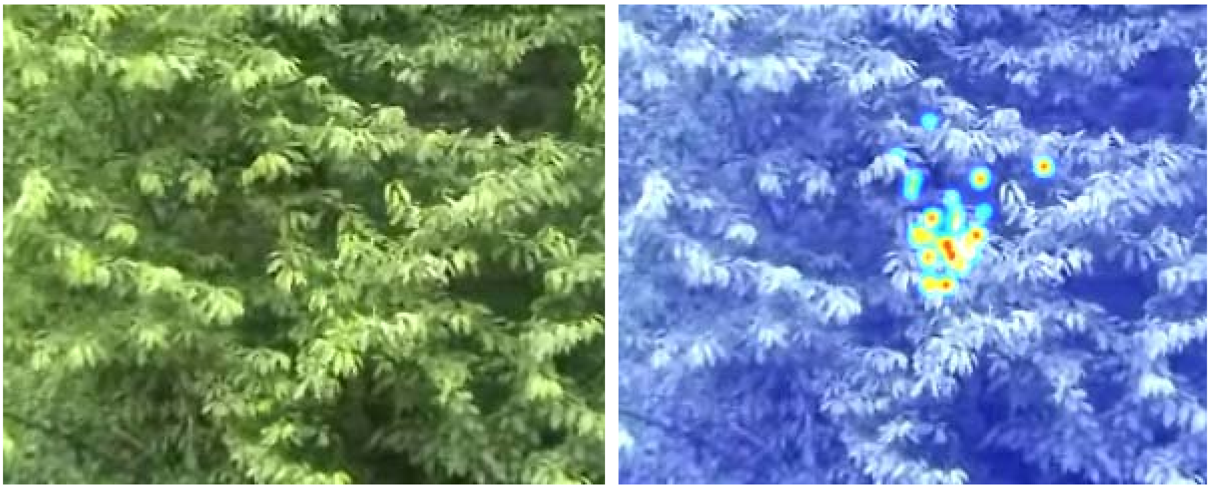
Salient Feature: A butterfly on the flower

06. Vol d'oiseaux .flv.MP4



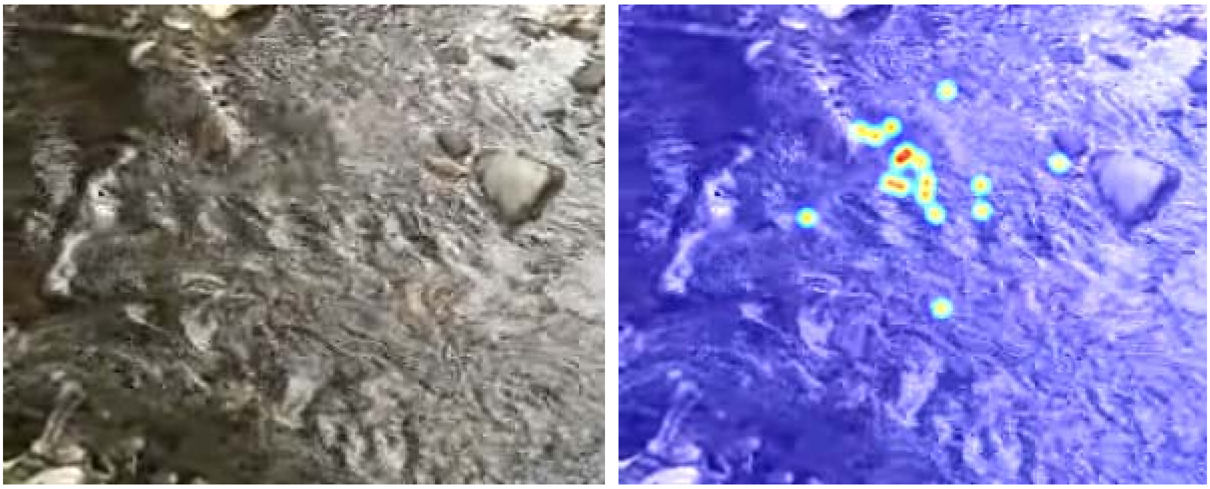
Salient feature: A bigger bird among many flying birds

07. cam3_new4.avi



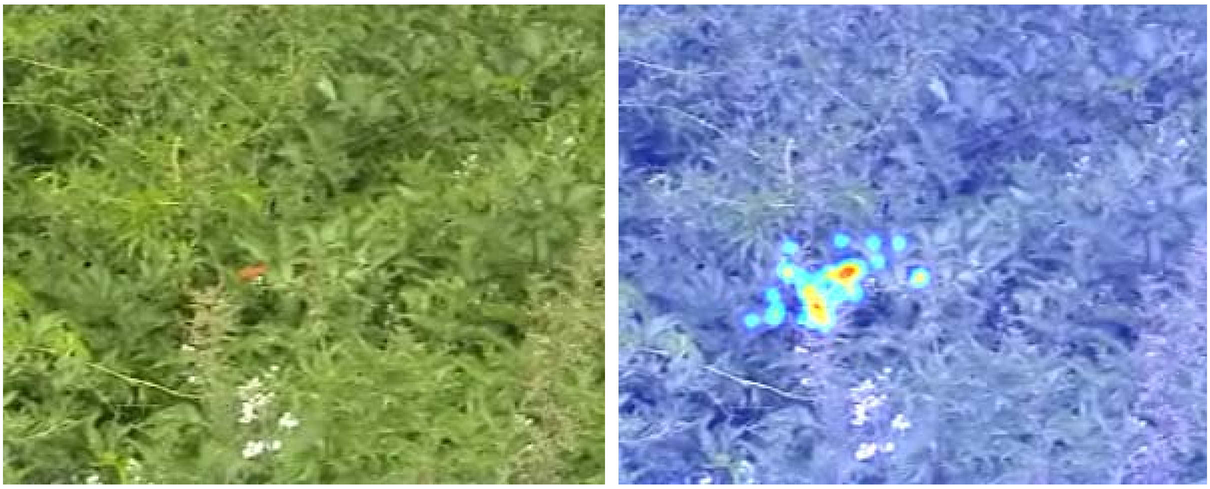
Salient feature: Zooming in of the scene

08. cam3_cam6.avi



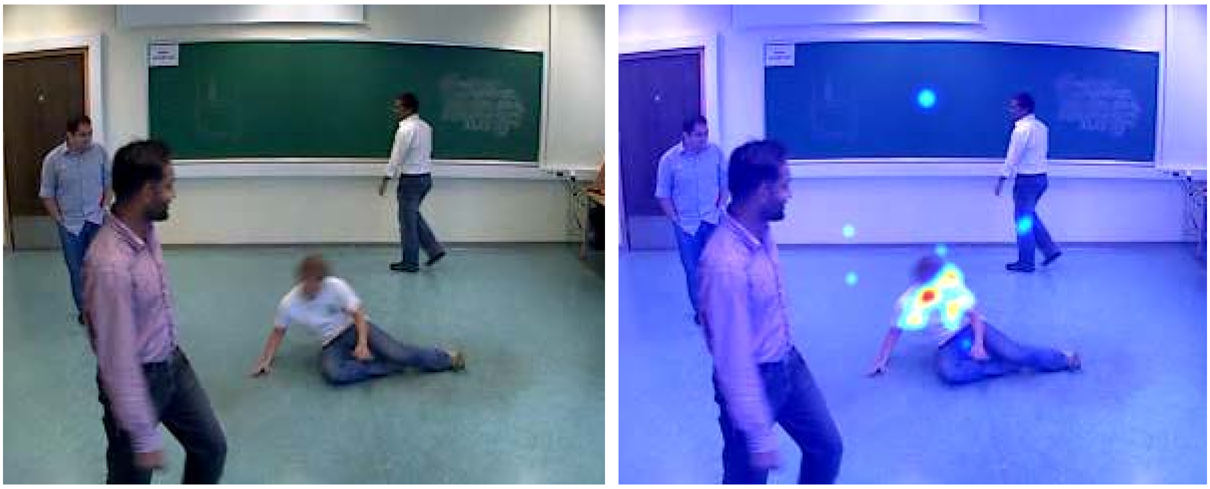
Salient feature: A stone dropped on the river

09. cam3_new1.avi



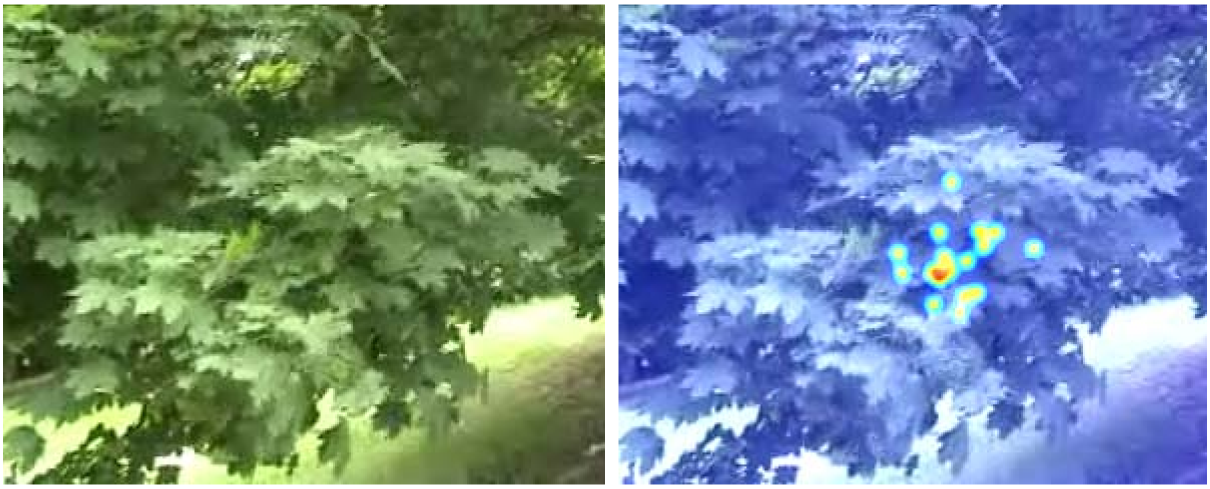
Salient feature: A hovering butterfly

10. AMC_Recording_part2.avi



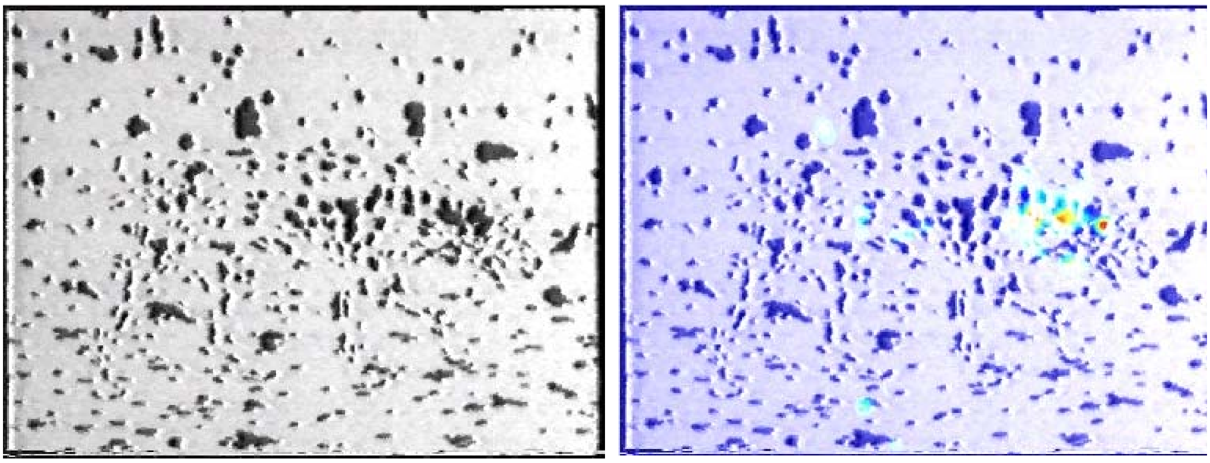
Salient feature: Yet another person falling down on the floor

11. cam3_tree.avi



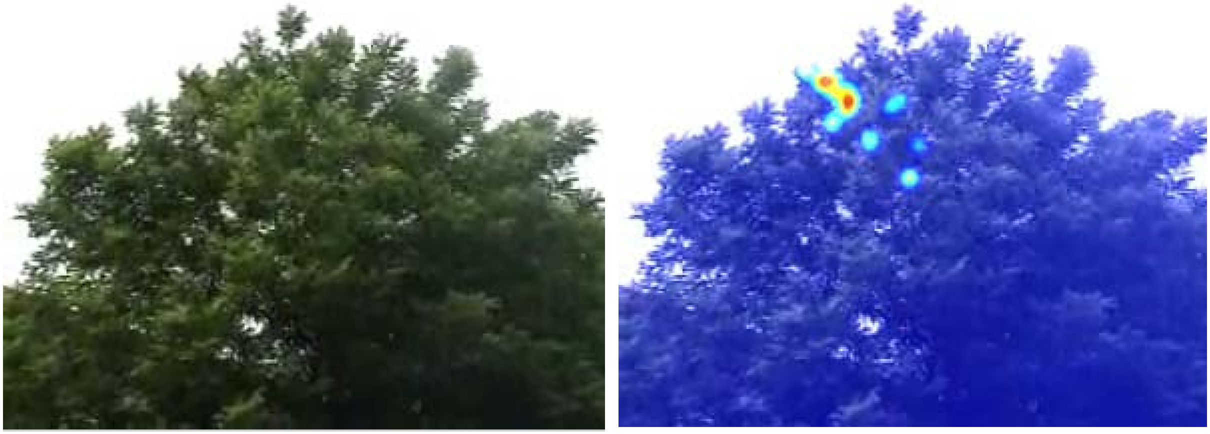
Salient feature: Moving leaves due to wind

12. DALMAT2.AVI



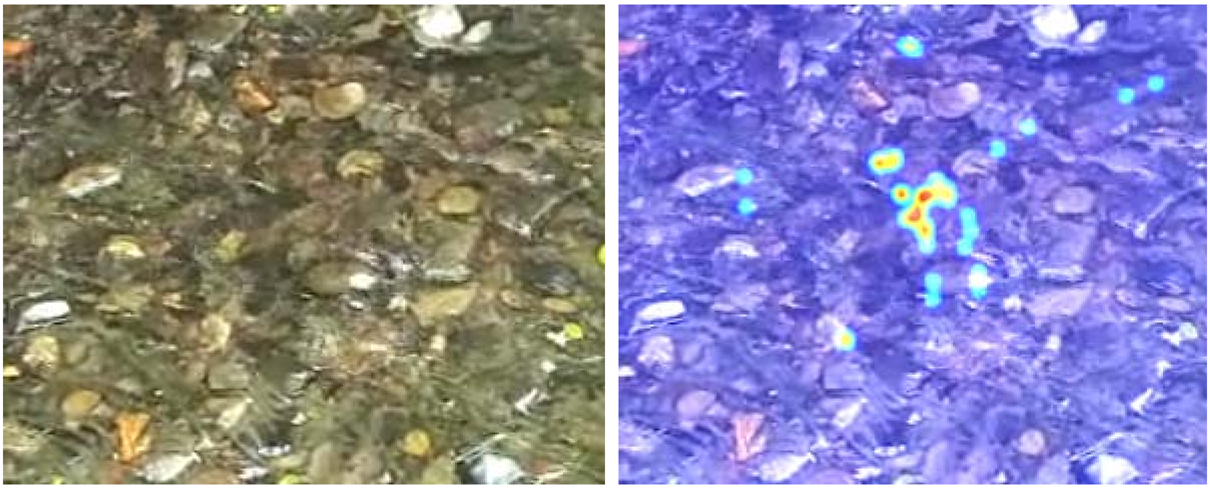
Salient feature: Moving dalmatian dog

13. tree2.wmv



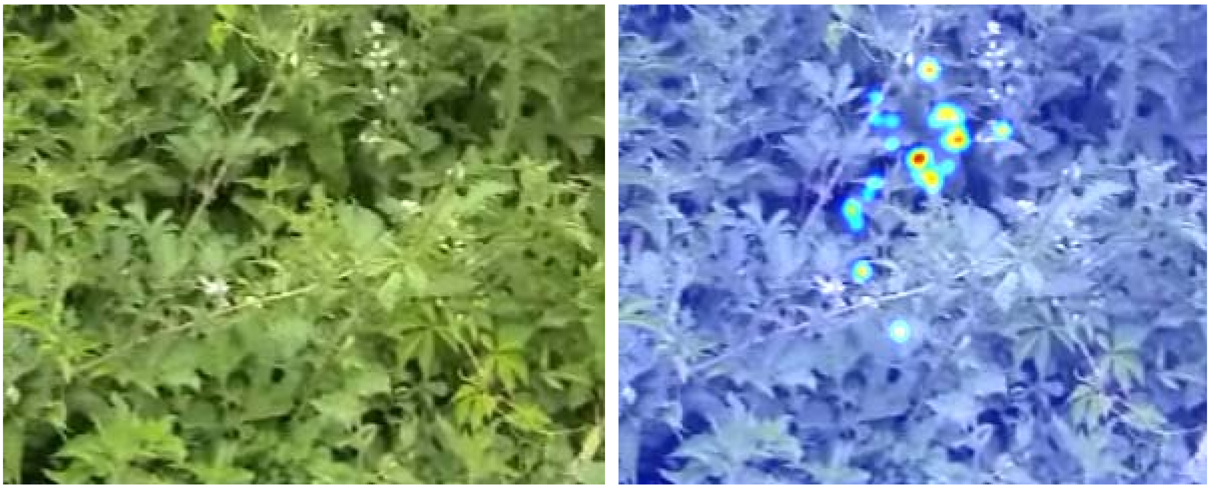
Salient feature: Something thrown in a parabolic path

14. cam3_new0.avi



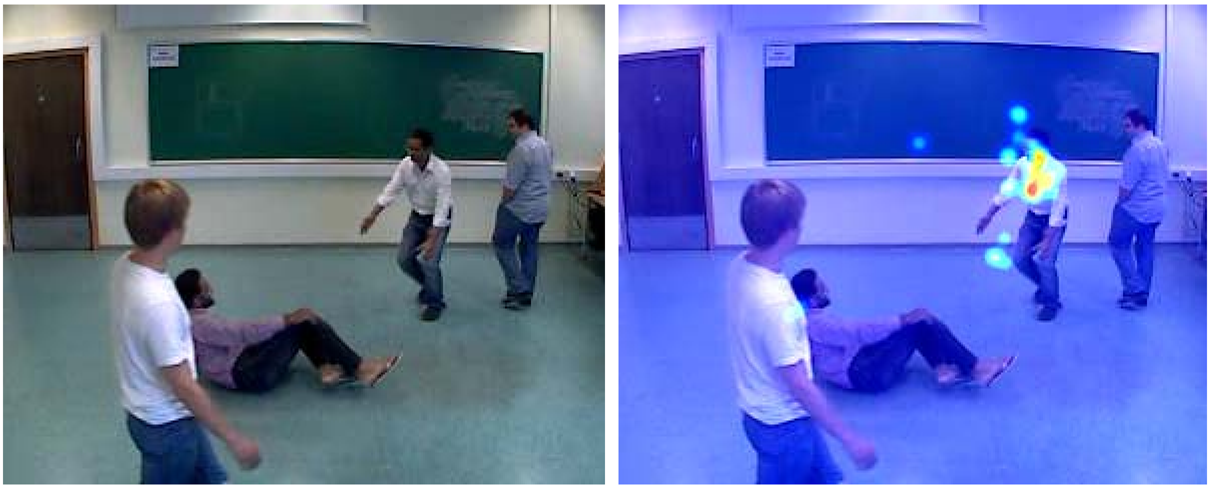
Salient feature: Flowing water on the river

15. cam22_new1.avi



Salient feature: Moving bushes due to wind

16. AMC_Recording_part3.avi



Salient feature: A person trying to rescue the falling person on the floor

Appendix C

Matlab Program Modules

The Matlab program files used and custom Matlab program modules developed for the experiment are described here. All the program modules and dependency files placed inside Matlab Code folder and they are provided along with this report.

The two Matlab program modules developed for the experiment are **RSRunExpt.m** and **RSGazeMap.m** are developed.

1. RSRunExpt.m : This program is used to execute the experiment. It runs the experiment either for gaze mapping image or video by displaying the images/videos sequentially and recording the gaze on the workstation computer.

```
Usage: RSRunExpt(inppath,outpath,inptype,validtypes)
Input Parameters:
   inppath: input path where the input files are located in stimulus pc
   outpath: output path where result data is to be saved in operator
            computer
   inptype: type of input; 'image' or 'video'
   validtypes: valid file types (list of acceptable file extensions
               like {'.bmp','.jpg','.gif'}
```

It will take the default value(s) if one or more parameters are not supplied. The default values of the parameters are as follows:

```
   inppath: . '\DATA\VIDEOS'
   outpath: 'D:\RS'
   inptype: 'video'
   validtypes: {'.avi','.wmv','.mp4'} – for inptype='video'
               {'.bmp','.tiff','.jpg','.gif','.png'} – for inptype='image'
```

2. RSGazeMap.m : This program module constructs the video gaze map from the original video and corresponding eye tracker data. This module reads the eye tracker data files in text format converted from IDF Converter and constructs gaze maped video from the original video and the corresponding gaze data

```
Usage: RSGazeMap(datapath,videopath,videotypes,gmappath)
Input Parameter:
   datapath: path containing all data from the eye tracker
   videopath: the path where the original videos to be used for the
              experiment are placed
   gmappath: the path where the resulting gaze mapped videos will be placed
```

It also takes the default value(s) if one or more parameters are not supplied. The default values of the parameters are as follows:

```
   datapath: 'DATA\ET_DATA'
   vidopath: 'DATA\VIDEOS'
   videotypes: {'.avi','.wmv','.mp4'}
   gmappath: 'DATA\VIDEOS\GMAPED'
```

3. RSPlayGazemapVideo.m : It can be used to play the gaze mapped videos in slow frame rate so that the gaze maps can be seen more clearly frame-by-frame. It allows to play the video in the rate as you wish by providing the frame rate parameter.

```
Usage: RSPlayGazemapVideos(videopath,videotypes,frate)
Input Parameter:
   videopath: the path where the gaze maped videos to be used for the
              experiment are placed
   videotypes: valid video types to be played in the format like
               {'.avi','.wmv'}
   frate: frame rate at which the videos are to be played in seconds (fps)
```

It takes the default value(s) if one or more parameters are not supplied. The default values of the parameters are as follows:

```
   vidopath: 'DATA\VIDEOS'
   videotypes: {'.avi','.wmv','.mp4'}
   frate: 8
```

Utility Program Files:

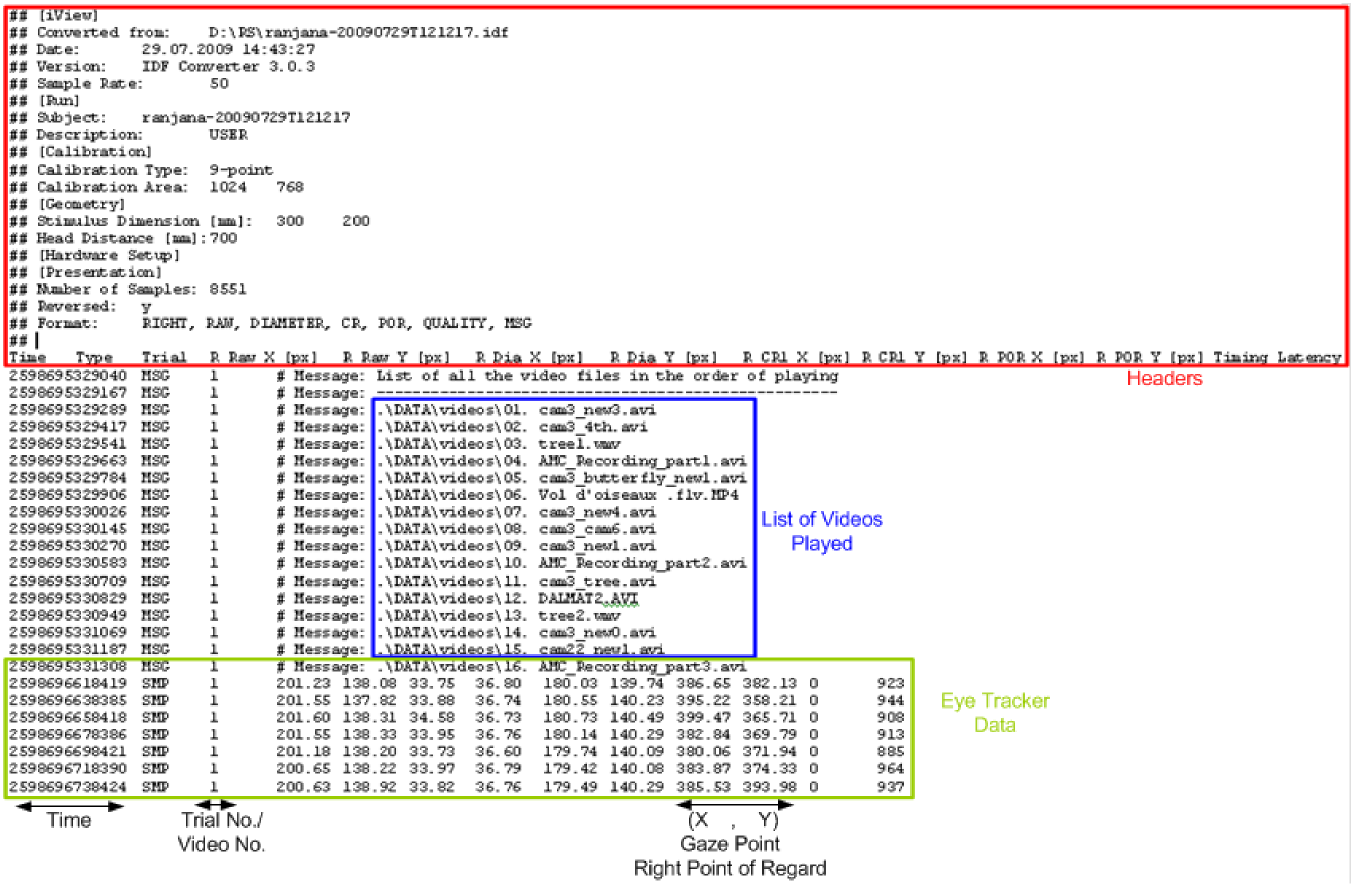
The utility program files that are used by the program are placed in the corresponding folders. The following list shows folders containing these dependent programs which are provided as a complete source programs along with this report. The program module or file in these folders which are directly called by the programs are mentioned here.

- **tcp_udp_ip:** It contains the files used for network connections between Stimulus PC and the Workstation PC. The **pnet()** function is used from these and either **pnet.dll** or **pnet.mexw32** file is the required file in this folder.
- **mmread:** It contains files for reading video files. The module directly called in the programs is **mmread.m**
- **mmwrite:** The **mmwrite()** function is called to write video into video files. For this function, either **mmwrite.dll** or **mmwrite.mexw32** file is required in this folder.
- **mmplay:** The function **mmplay()** is used to play videos and it requires either **mmplay.dll** or **mmplay.mexw32** file in this folder.
- **grayscaleops:** This folder contains gray scale operation files. **graydil.m** Matlab program module is used for grayscale dilation on images.

Appendix D

Eye Tracker Data Format

The following format shows the format of the eye tracker data in text file converted from .idf data file using IDF Converter tool. The main areas of the data format and useful data fields are labeled below.



```
## [iView]
## Converted from:     D:\RS\ranjana-20090729T121217.idf
## Date:        29.07.2009 14:43:27
## Version:     IDF Converter 3.0.3
## Sample Rate:         50
## [Run]
## Subject:     ranjana-20090729T121217
## Description:         USER
## [Calibration]
## Calibration Type:   9-point
## Calibration Area:   1024    768
## [Geometry]
## Stimulus Dimension [mm]:   300     200
## Head Distance [mm]:700
## [Hardware Setup]
## [Presentation]
## Number of Samples:  8551
## Reversed:   y
## Format:      RIGHT, RAW, DIAMETER, CR, POR, QUALITY, MSG
## |
Time    Type    Trial   R Raw X [px]    R Raw Y [px]    R Dia X [px]    R Dia Y [px]    R CR1 X [px] R CR1 Y [px] R POR X [px] R POR Y [px] Timing Latency
2598695329040 MSG     1         # Message: List of all the video files in the order of playing
2598695329167 MSG     1         # Message: ------------------------------------------------
2598695329289 MSG     1         # Message: .\DATA\videos\01.  cam3_new3.avi
2598695329417 MSG     1         # Message: .\DATA\videos\02.  cam3_4th.avi
2598695329541 MSG     1         # Message: .\DATA\videos\03.  tree1.wmv
2598695329663 MSG     1         # Message: .\DATA\videos\04.  AMC_Recording_part1.avi
2598695329784 MSG     1         # Message: .\DATA\videos\05.  cam3_butterfly_new1.avi
2598695329906 MSG     1         # Message: .\DATA\videos\06.  Vol d'oiseaux .flv.MP4
2598695330026 MSG     1         # Message: .\DATA\videos\07.  cam3_new4.avi
2598695330145 MSG     1         # Message: .\DATA\videos\08.  cam3_cam6.avi
2598695330270 MSG     1         # Message: .\DATA\videos\09.  cam3_new1.avi
2598695330583 MSG     1         # Message: .\DATA\videos\10.  AMC_Recording_part2.avi
2598695330709 MSG     1         # Message: .\DATA\videos\11.  cam3_tree.avi
2598695330829 MSG     1         # Message: .\DATA\videos\12.  DALMAT2.AVI
2598695330949 MSG     1         # Message: .\DATA\videos\13.  tree2.wmv
2598695331069 MSG     1         # Message: .\DATA\videos\14.  cam3_new0.avi
2598695331187 MSG     1         # Message: .\DATA\videos\15.  cam22_new1.avi
2598695331308 MSG     1         # Message: .\DATA\videos\16.  AMC_Recording_part3.avi
2598696618419 SMP     1         201.23  138.08  33.75   36.80   180.03  139.74  386.65  382.13  0       923
2598696638385 SMP     1         201.55  137.82  33.88   36.74   180.55  140.23  395.22  358.21  0       944
2598696658418 SMP     1         201.60  138.31  34.58   36.73   180.73  140.49  399.47  365.71  0       908
2598696678386 SMP     1         201.55  138.33  33.95   36.76   180.14  140.29  382.84  369.79  0       913
2598696698421 SMP     1         201.18  138.20  33.73   36.60   179.74  140.09  380.06  371.94  0       885
2598696718390 SMP     1         200.65  138.22  33.97   36.79   179.42  140.08  383.87  374.33  0       964
2598696738424 SMP     1         200.63  138.92  33.82   36.76   179.49  140.29  385.53  393.98  0       937
```