User Manual for the SFU Eye-Tracking Database

Version 1.1, Feb. 2011

Multimedia Communications Laboratory
School of Engineering Science
Simon Fraser University

Eye-Tracking Database User Manual

Written by Hadi Hadizadeh (hha54@sfu.ca), Oct. 2010-Feb. 2011

Thank you for downloading this database. The database files are organized as follows:

A. Database Location, Structure and Accessibility

The database is available online at the following URL:
www.sfu.ca/~ibajic/datasets.html.

B. Video Sequences

To generate the eye-tracking data, we used the following 12 standard video sequences: *Foreman* (300 frames), *Bus* (150 frames), *City* (300 frames), *Crew* (300 frames), *Flower Garden* (250 frames), *Mother and Daughter* (300 frames), *Soccer* (300 frames), *Stefan* (90 frames), *Mobile Calendar* (300 frames), *Harbor* (300 frames), and *Tempete* (260 frames). The sequences are stored in YUV 4:2:0 format at CIF (352×288) resolution, and 30 frames per second (fps). The video sequences (in uncompressed YUV 4:2:0 format) are stored in the "RAW" folder.

C. Eye-Tracking Visualization Videos

A total of 15 non-expert participants (2 women and 13 men, aged between 18 and 30) took part in the eye-tracking data collection study. The study was performed one participant at a time in a quiet room with an ambient light of 200 Lux.

The 12 short video sequences were presented sequentially in a fixed order with a 3 second pause in-between. During this pause and before the beginning of each video, a small crosshair (centered on the video display area) was presented and the participants asked to fixate on it. After the 12 videos had been presented, participants then had a 2-minute break after which the 12 videos were presented again. The participants were asked

to look naturally at the videos and were not given any instructions as to what to look for in the sequences. The gaze data was represented in two different visualizations for each video sequence: a moving heat map and a gaze plot comparing participants' first and second viewing of the sequences. In the heat map visualization, the areas of the video that received the most visual attention are presented in white, followed by red, yellow green and blue as visual attention dropped. In the gaze plot visualization, a pair of connected circles represent where each participant looked at the sequences the first and second time they were presented to them. Each participant's gaze location for the first and second view is represented in a different color.

The visualization video files are stored in the "Visualizations" folder and can be played using either Windows™ Media Player or Quicktime™. There are two different visualization videos for each of the sequences:

**_1vs2View.avi* : this visualization represents gaze locations comparing first vs. second viewing of the video for each of the 15 participants (joined circles with different colors for each participant);

**_Heatmap.avi* : This visualization represents a moving heat map visualization for each of the video sequences.

D. Gaze Data Files

The collected raw gaze data was analyzed, mapped from the head-mounted eye tracker onto the video plane and stored in a comma separated value (CSV) file format. This file contains the frame-by-frame, pixel wise *x*- and *y*-coordinates (measured from the bottom left corner) of the gaze location for each participant and each of the video sequences.

The Gaze data files can be found in the "CSV" folder and can be opened with any spreadsheet software. The format of the file is as follows: the first and second rows contain file descriptions and headers, the remaining rows contain the X and Y pixel coordinates for each of the 2 viewings by the 15 participants for every frame. Each frame is represented by a row. In this manner, row 3 in the file contains the gaze locations for the first frame of the video, row 4 contains frame 2 and so on.

E. MATLAB code

The following three MATLAB programs are provided in order to display gaze locations and generate heat maps based on the gaze data files:

- *demo_gaze_locations.m* : displays gaze locations on each frame of a given video sequence. This demo program calls function *ShowGazeLocations* which display the gaze locations on each frame of the input YUV video sequence based on the input gaze data file (in CSV format), and the selected viewing (first viewing, second viewing, or both). Type "*help ShowGazeLocations*" on the MATLAB

command line to see the description of each input parameter of *ShowGazeLocations*.

- *demo_heatmaps.m* : displays heatmaps on each frame of a given video sequence. This demo program calls function *GenerateHeatMaps* which display a heat map on each frame of the input YUV video sequence based on the input gaze data file (in CSV format), and the selected viewing (first viewing or second viewing). Type "*help GenerateHeatMaps*" on the MATLAB command line to see the description of each input parameter of *GenerateHeatMaps*.

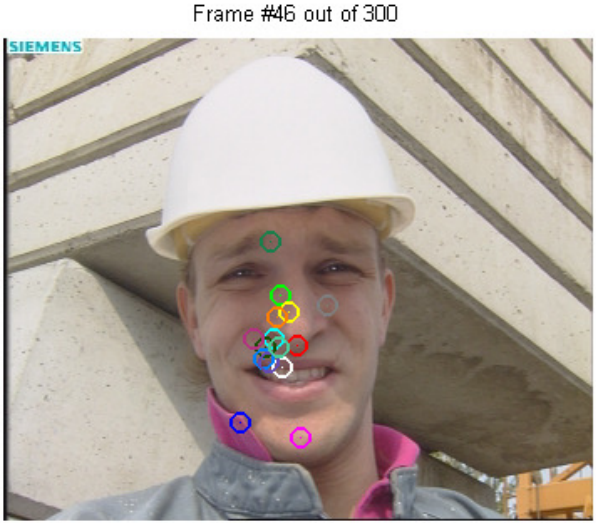
- *GenerateMask.m* : generates a binary mask for the given input CSV file containing the gaze data for the relevant sequence. This function automatically flags those invalid data points (outliers) in our database using the following automated procedure: This procedure flags as invalid those gaze locations that satisfy the following conditions:

  - The gaze location is out of frame boundaries;
  - The gaze location is at the frame boundaries or very close to the frame boundaries (less than 5 pixels);
  - The gaze location remains constant for N consecutive frames. The value of N can be set arbitrarily by the user. In our current database, we set $N = 30$ frames (i.e., 1 sec);
  - The gaze location remains constant in all frames.

  Using this procedure, a binary flag matrix is generated for each CSV file in the database that contains the gaze data. A zero entry indicates that the corresponding gaze location in the relevant sequence or viewing (for the related subject) is invalid. Both *ShowGazeLocations* and *GenerateHeatMaps* employ this function to discard invalid data. Type "*help GenerateMask*" on the MATLAB command line to see the description of each input parameter of *GenerateMask*.

- *EWPSNR.m* : computes an eyetracking-weighted PSNR (for the definition, see Z. Li, S. Qin, and L. Itti, "Visual attention guided bit allocation in video compression," *Image and Vision Computing*, vol. 29, no. 1, pp. 1-14, Jan. 2011.) between two frames of two input YUV 4:2:0 videos, given the location of the fixation point in the frame. Type "*help EWPNSR.m*" in the MATLAB command line to see the description of each input parameter of *EWPSNR.m*.

An example of the heat map for *Foreman* generated by *GenerateHeatMaps*



An example of the visualization of the gaze locations for *Foreman* generated by *ShowGazeLocations* (only the first viewing is shown)

Contact Us!

For more information, feel free to contact us:

Ivan V. Bajić: ibajic@ensc.sfu.ca
Hadi Hadizadeh: hha54@sfu.ca
Mario J. Enriquez: mario_enriquez@sfu.ca

Multimedia Communications Laboratory
School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, BC, V5A 1S6, Canada