Offline tracking of eyes and more with a simple webcam

Bo Pedersen (bop@cornell.edu)
Michael Spivey (spivey@cornell.edu)
Department of Psychology, Cornell University
Uris Hall, Ithaca, NY 14853 USA

Abstract

The quality of modern day webcams allow them to be used for eye-tracking and a substantial amount of research is being dedicated to the development of fast and robust software that will allow users to navigate their computers using webcams only. With these demands such software is bound to become very complicated, but we will demonstrate that this is only because of speed and robustness constraints. For most experiments in Cognitive Science it is sufficient to analyze recorded webcam videos offline and this dramatically reduces the problem since conceptually simple methods that are normally abandoned because they are too slow, can be used, since time is not an issue, and problems related to calibration and robustness can be dealt with in the lab after the subject has left. In one experiment we show that tracking points calculated from a webcam recording of the eye are highly correlated with tracking points from a head-mounted eye-tracker on the horizontal axis and in a second experiment we demonstrate that webcams can be successfully used for a couple of classic eye-tracking tasks and finally that the richness of the data allow for extraction of more than just eye movements.

When we casually observe other humans' eye movements we can often reliably infer what they are looking at based on the set of possible landmarks in the scene in front of them (Gibson & Pick , 1963). This can be modeled by having a set of images of the eye and a corresponding set of x and y coordinates, and whenever we have a certain picture of the eye we can produce a set of coordinates, and if we make the model continuous we can get coordinates for novel eye images based on their similarity with the existing images. Therefore, when recording a subject's eye, we need to make sure that some of the images have corresponding coordinates. Traditionally we refer to these as calibration points, but since the analysis of the video is offline there are less constraints on these points. They can in fact be positioned after the main stimuli and one could even speculate in ways to eliminate them entirely for example by relying on points within the stimuli.

Systems that use inexpensive webcams for eye-tracking exist but the focus is most often on fast online tracking that the disabled can use for communication with eye-typing software (See the COGAIN initiative, cogain.org and Hansen et al, 2001). Central to the current method is that it is offline and use examples of the current subjects eye-positions to estimate novel eye-positions and movements.

A prerequisite for this method is to find the eye within the picture. Since we are doing the analysis offline we could in fact just mark these up manually before starting the analysis. However, for the present data the eyes have been found automatically within the picture by accumulating the frame-by-frame difference, and then find the horizontal line with the most difference over all frames. This line will usually cut through the eyes and then on this line we can find two local maxima corresponding to the horizontal position of the eyes. Figure 1 show this process with the vertical maximum to the left of the picture and the horizontal maximum under the picture.

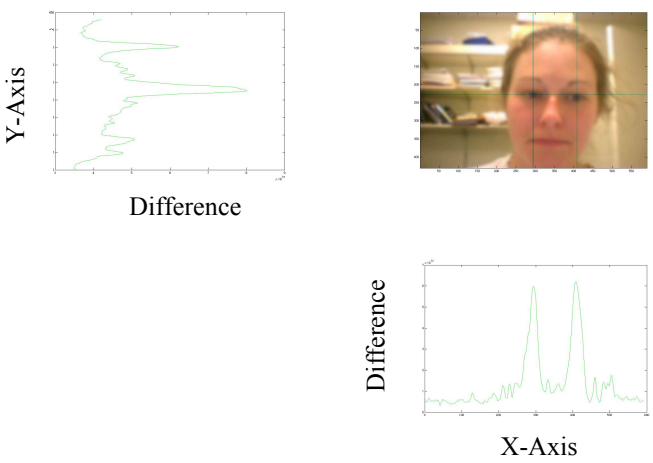

Figure 1 - Finding the eyes

Having found the eyes, we collect the N calibration scene images and create an NxN similarity table based on the error of trying to fit each of the calibration eye images into each of the calibration scene images point by point. So we can now produce x,y-coordinates from a row of N numbers in this table. This is useful for interpolation. For every new eye image, we can now fit all N calibration eye images into the scene and get a new row of N errors that can be compared to the other rows. To estimate the x,y-coordinates for novel eye-images we train a neural network on the relation between the relation between the calibration images and the calibration points. It is worth adding that the point-by-point comparison of images is a computationally intense process that would be totally useless if time were an issue, but since

this is an offline analysis we have all the time in the world. Figure 2 shows what this looks like for 3 calibration points.

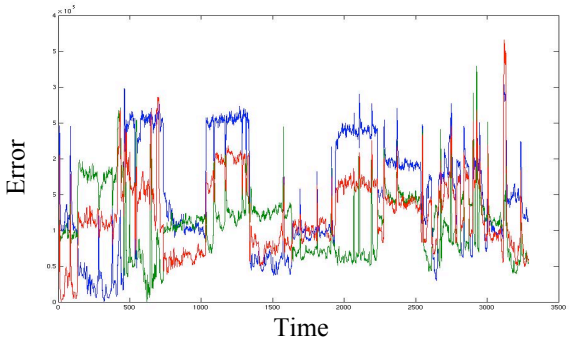

Figure 2 - Errors for fitting 3 different prototypes

Experiment 1

A pilot experiment was conducted where recordings of both eye-tracker data and webcam data were collected in order to compare. An ISCAN headband-mounted eye-tracker and an iSight webcam were used. The subject was calibrated on 9 (3x3) points and watched 2 minutes of a Martha Stewart cooking video. Eye-blinks were removed from the data and a simple regression was calculated for horizontal points and for vertical points across the two datasets. Horizontally the R^2 was .90 and vertically .26. ($p<0.001$) Figure 3 shows the scatter plots of the horizontal and vertical conditions. The correlations are made under the assumption that they are linear but it looks like the horizontal correlation might involve a curve-linear fit.

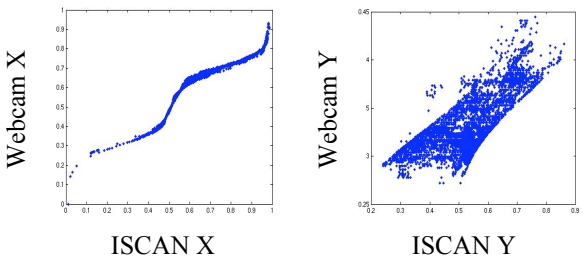

Figure 3 - Scatter plots for horizontal and vertical points

Experiment 2

5 Subjects were instructed to sit down and watch 4 minutes of stimuli while being recorded with a webcam and they were told that the first screen would explain what to do in the experiment. The first screen consist of 9 lines thanking the subjects for participating in the experiment and informing them that they will be reading some text and watch some images but besides that do nothing, except for the final calibration procedure where we want them to keep looking at the numbers presented on the screen. 5 images were then presented one at a time with 10 seconds for each image. The first image is the painting used in Yarbus (1967) "They did not expect him". The following 4 images were cartoons from wulfmorgenthaler.com. An extensive post-calibration where the numbers between 1 and 25 were then presented to the subjects in 4 different orders and with different speeds in order to get the calibration points in as many different contexts as possible. Figure 4 shows an example of one subject's eye taken from the first of the 4 calibration sequences.

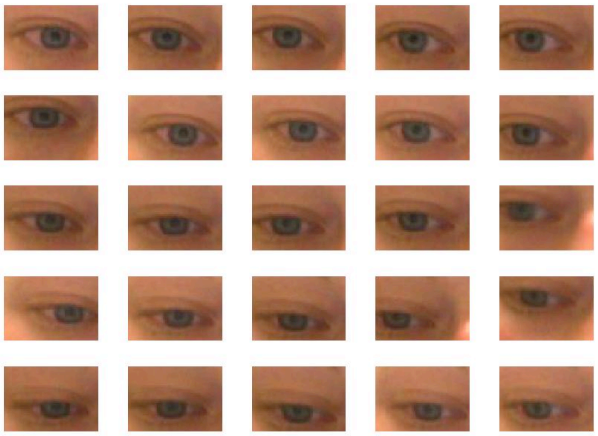

Figure 4 - Images of the eye at the 25 calibration points

The first calibration block used 4 seconds for each point and the next 2 seconds, 1 second and .5 second in easier and easier patterns (First in steps of 8 in the 5x5 grid and then 4, 2 and 1). The first block turns out to be enough for the calibration so the consecutive blocks can be used as measures of how successful the calibration is. Regressing from block 2 to block 1 gives an R^2 of .87 horizontally and .47 vertically on average across all subjects over a series of 30 points in each block. ($p<0.001$). This is just a rough average that doesn't take out blinks and other noise. Given that the scan paths are different for block 1 and block 2, this is a reasonable result considering that the subjects were not using a chin-rest nor were they instructed to sit still. They were in fact moving around a bit and touching their face, etc.

In figure 5 we have plotted the error of the upper left prototype during the first 25 seconds where the subject is reading the instructions on the screen.

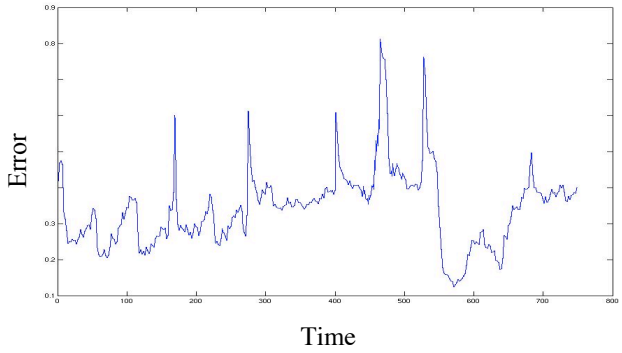

Figure 5 - Error of a prototype during reading

A handful of interesting things can be read directly from this graph. The spikes are the blinks of the subject and we can see that there about 5 of these. These can be detected very precisely by summing over the error of all 25 prototypes. It can also be seen that the there are a number of local gradual increases and sudden decreases that directly reflect that the subject is reading and the global increase in 2/3's of the graph reflect that the subject is moving down the page.

Saccades, and therefore also fixations, are also reasonably detectable. Figure 6 shows the frame-by-frame differences between error-terms for the first 3 seconds of reading:

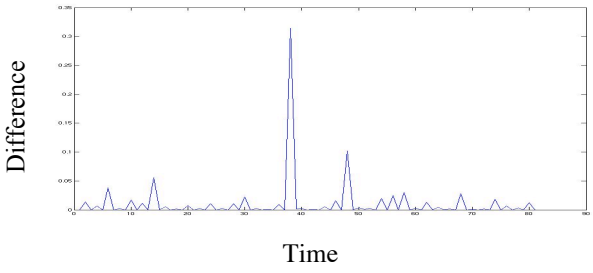

Figure 6 - Error difference indicating saccades

Some small spikes reflecting the word-by-word saccades can be seen and a big one reflecting the jump to the next line.

Finally, in figure 7 we graph the saccades and fixations for the first 3 seconds where the subject is reading the first 2 lines.

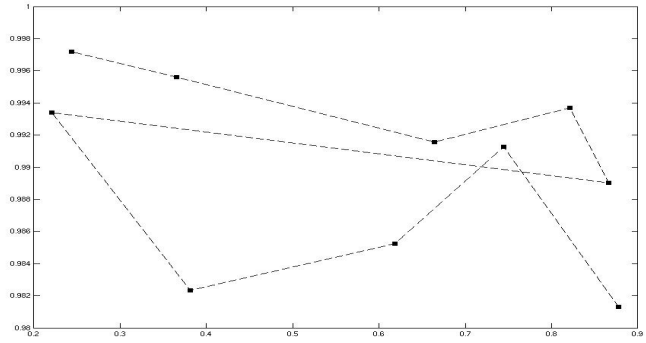

Figure 7 - Saccades and fixations during reading

As expected, the vertical acuity is not very good. The method might still be useful for coarse analyses of reading performance in HCI tasks since we know were the lines are and we know that the subject is likely to read line 2 after line 1. However, this method is not yet ready for fine-grained analysis of fixation patterns of individual words during sentence processing tasks (Rayner, 1998).

Say Cheese

Since we are constantly finding the minimum errors for fitting the calibration eye images into the picture, we also know where this minimum error is found, so we do in fact automatically have a good measure of where the head is located in the picture. This might be useful for various purposes and at the very least it can be used to correct the predicted x,y-coordinates. Since we can predict where the eye is looking, it is likely that we can predict where the face is turning, using exactly the same techniques. So it seems that we can extract a whole battery of continuous variables. As an example of the open nature of this list of variables, imagine that we want to test if the subjects find certain cartoons funny or not. Then we might want to see to what extent they are smiling during these stimuli (the variation in head-movement caused by laughter might be an additional measure for this). In the stimuli, we did in fact have 4 cartoons and one control image and by the end of the calibration we also asked them to smile so we have an independent measure of their smile. We used this image and a control image taken right before the calibration smile to calculate a running error. From both images, a region below the right eye including a bit of the nose, mouth and chin was selected. Figure 8 shows the error of fitting the control image *minus* the error of fitting the calibration smile image, so high values should correspond to smiles. The graph is a smoothed, normalized average of 5 subjects.

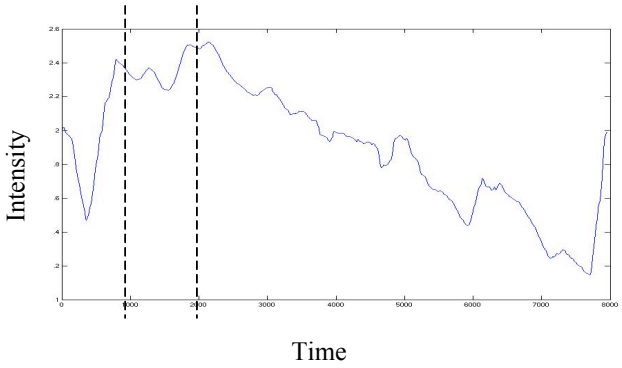

Figure 8 - Smile intensity

As can be seen there is a region with 4 little maxima between 1000 and 2000 in the graph corresponding to the 4 cartoons after the introduction screen and the control image. Afterwards there is a gradual decrease until the onset of the calibration smile. A T-test on the difference between points in the 1000-2000 interval and the rest of the points is significant ($p<0.001$).

More systematic methods for "Automatic Facial Expression Analysis" are available (see Cohn & Kanade, in press). But as with the eye-tracking methods, these are very complex since they try to be fast online methods that work directly across all subjects, where the present method is very simple because it use prototypes from the actual subject.

Conclusion

The first software *products* for doing eye-tracking with webcams have already hit the market, and more are to follow, but given the resources that have been put into the development of them they are bound to be expensive and proprietary, platform-dependent, black-box and probably hard to integrate into any cognitive science experiment. We have here shown that because analysis of the data can be done offline for most experiments, very simple techniques can be used to extract eye-tracking data and we offer a free, open and intelligible alternative, much in the spirit of the open source community - and we will in fact share the code - but we would rather think of it as sharing an idea. In fact the first version of the code was only 20 lines of matlab code that any scholar would be able to comprehend and modify. Of course, the devil is in the details, and it is important that scientists know their tools intimately – especially when studying something as complex as the real-time dynamics of the mind. There is a long history of painstaking techniques for recording eye position (for review, see Richardson & Spivey, 2004a), and eye-tracking has been used to study an extremely wide variety of cognitive processes (for reviews, see Rayner, 1998; Richardson & Spivey, 2004b; Underwood, 2005). With the small sacrifice of some precision, the ease of use with the present method, and its wide range of applications to related facial movements, makes it an excellent candidate for further exploration.

References

Cohn, J. F. & Kanade, T. (In press). *Use of automated facial image analysis for measurement of emotion expression*. In J. A. Coan & J. B. Allen (Eds.), The handbook of emotion elicitation and assessment. Oxford University Press Series in Affective Science. New York: Oxford.

Gibson, J.J. & Pick, A.D. (1963). Perception of another person's looking behavior. *American Journal of Psychology, 76*, 386-394.

Hansen, P.J., Hansen, D.W. & Johansen A.S. (2001). *Bringing Gaze-based Interaction Back to Basics*, in Proceedings of Universal Access in Human-Computer Interaction (UAHCI 2001), New Orleans, Louisiana.

Rayner, K. Eye movements in reading and information processing: 20 years of research. Psychological Bulletin 124, 3 (1998), 372--422.

Richardson, D.C. & Spivey, M.J. (2004). *Eye tracking: Characteristics and methods*. In G.Wnek & G.Bowlin (Eds.), Encyclopedia of biomaterials and biomedical engineering (pp. 568–572). New York: Marcel Dekker.

Richardson, D.C. & Spivey, M.J. (2004). *Eye tracking: Research areas and applications*. In G.Wnek & G.Bowlin (Eds.), Encyclopedia of biomaterials and biomedical engineering (pp. 573–582). New York: Marcel Dekker.

Underwood, G. (2005). *Cognitive Processes in Eye Guidance*. Oxford University Press, UK.

Yarbus A. (1967) *Eye movements during perception of complex objects*. In L. A. Riggs, Ed., Eye Movements and Vision, Plenum Press, New York, Chapter VII, 171-196