Comparison of eye movement filters used in HCI

Oleg Špakov
Tampere Human Computer Interaction Unit
School of Information Sciences
University of Tampere
oleg.spakov@uta.fi

Abstract

The study compares various real-time filters designed to denoise eye movements from low-sampling devices using four criteria. Most of the filters found in literature were implemented and tested on data gathered in a previous study. An improvement was proposed for one of the filters. Parameters of each filter were adjusted to ensure their best performance. The output from the filters was compared against two idealized signals (the signals denoised offline). The study revealed that FIR filters with triangular or Gaussian kernel (weighting) functions and parameters dependent on signal state show the best performance.

CR Categories: H.5.2 [Information Interfaces and Presentation]: User Interfaces – Evaluation/methodology; Input devices and strategies

Keywords: eye tracking, gaze, filters, smoothing, algorithms

1 Introduction

Eye movements measured with eye-tracking systems commonly used for HCI purposes always contain some noise due to imperfection of the measuring technology. It is common to treat eye movement as a signal with states: saccadic (ballistic jumps) and fixational ("still" eye). However, fixations consist of micro-movements and tremor, which have amplitudes of about same order as noise. Both noise and actual eye movements during fixations are the unwanted features of the signal used as input for HCI interface, therefore the first step taken when mapping gaze onto onscreen objects is raw data filtering and smoothing.

Despite this problem was defines already in early 1980s [e.g., Tole and Young 1981], filtering of data samples with low frequency (~30-100 Hz) was not sufficiently studied yet. Many authors make various recommendations about filters and suggest different algorithms to be used, but no deep analysis of existing algorithms was performed so far. Our study aims to fill this gap in knowledge. In this paper we first present a wide range of filters found in literature, then selected for comparison those we find suitable for this kind of study, propose suggestions for improvement of some of them, define comparison criteria, and finally present the results and discuss them.

2 Background

The simplest filtering of gaze data consists of substituting the received gaze point by a simple average of the last N points (including the one that is being replaced by this average). This kind of data smoothing was used mostly in early studies (e.g., [Jacob 1993]), but may be found in recent works as well (e.g., [Zhang et al. 2010]). The strongest disadvantage of this filter is the introduced delay in update of the gaze point position after saccade: the first few (usually, 1-3) gaze points of a new fixation have too little share (weight) in the averaging (usually, over 4-7 points) to quickly respond to a strong change of the input signal.

To overcome this issue, various finite-impulse response (FIR) filters were employed in gaze data smoothing. These filters use buffer filled by the latest gaze points, and each point has its own weight when calculating output as a weighted average. The function that assigns weights W_i to each point is called *the kernel function*, although in some early works a manually set weights were proposed instead (e.g., in [Duchowski 2003]). If a kernel function assigns "1" to each weight, then a FIR filter outputs a usual average of N samples, and the kernel function is called linear. At least two other kernel functions used in FIR filters were described in literature: *triangular* (e.g., [Kumar et al. 2008]) and *Gaussian* (e.g., [Jimenez et al. 2008). The triangular kernel function assigns "1" to the least recent gaze point ($W_N = 1$), "2" to the next point ($W_{N-1} = 2$), and so on ($W_i = N - i + 1$). The Gaussian kernel function is expressed as follows:

$$W_i = e^{-\frac{(i-1)^2}{2\sigma^2}}$$

When calculating average with weights set by the Gaussian kernel function, the points which get $W < 0.05$ are usually ignored; in other words, the length of the buffer is selected so that each point in it has $W_i \geq 0.05$. The following formula is used to calculate the output value from a FIR filter:

$$FP = \frac{\sum_{i=1}^{N} W_i * RP_i}{\sum_{i=1}^{N} W_i}$$

where FP is the output point, and RP is a point of the input signal. The weighted averaging alone strongly improves filter response (decreases delay) to sudden changes in gaze point location. However, a signal state discrimination and other additional algorithms are often used in FIR filters to archive virtually delay-free performance. For example, Kumar et al. [2008] used outlier and saccade detectors to manage the number of the gaze points in the averaging buffer. Saccade detector is a simple comparer of the location of the current input point and the last output point against a predefined threshold (distance). When the threshold is exceeded, the filter buffer is cleared. In some studies this detector was improved by an algorithm that obtains the threshold dynamically according to the signal state [Tole 1981; Duchowski 2003].

The outlier detector postpones the update of the output point if the current gaze point lies far from it. This way single outliers can be detected and ignored; however, this detector introduces a 1-point delay when saccade starts, but it may be considered as not a crucial disadvantage if the sampling frequency is relatively high (>100 Hz). More robust outlier detectors (2+ outlying points) may introduce greater delays: see, for example, the two-leveled peak removing algorithm described by Stampe [1993].

Another method to manage the length of buffer is described in van de Kamp and Sundstedt [2011]. This method makes a buffer small (few points) immediately after a saccade detection, and gradually increases it during fixations. Similar approach was taken by Veneri et al. [2010]: the buffer is cleared after saccade detector, and gradually grows during fixation until its desired size is reached. Veneri et al. [2010] used simple averaging.

FIR filters may be using in chain with other filters which may or may not affect FIR filter parameters. For example, in the work by Gu et al. [2000] the least square optimal filters were preceding the FIR filter. This works is interesting also due to the median function used in calculating the output points, whereas mean function was used in all other studies.

The low-pass filter designed by Olsson [2007] had two functional states: strict (during fixations) and soft (during saccades). The state detection algorithm, as in other solutions, used a spatial threshold to separate slow and fast eye movements. However, the distance was measured between 2 sets of gaze points rather than between latest input and previous output points.

Komogortsev and Khan [2007] reported about successfully applied a Kalman filter for smoothing gaze path. The filter used gaze position and its movement speed as the estimation targets.

Finally, Sesin et al. [2008] described an artificial neural network trained to filter noise from eye movements. The network had 12 inputs (6 recent gaze points) and one hidden layer of 24 nodes (neurons). Authors claimed they could successfully train the network within 3 minutes and archive filtering good enough for the purpose of their study.

Not all filters observed here were selected for comparison, although most were implemented and tested to some extend. Some filters were found to be slow in response to saccades (like filters without signal state detection), others showed poor smoothing capabilities (like the artificial neural network) despite careful implementation according to how they were described in literature. On the other hand, all FIR filters were tested with each of three kernel functions, although they were designed and tested using only one of them in the original works. In addition, a Savitzky-Golay filter used by Nyström and Holmqvist [2010] for offline smoothing was included into the comparison as it is possible to tune its parameters to have only 1-point delay. See the original publication for the detailed description of each filter. All filters selected for comparison are listed in Table 1.

Table 1 *Tested filters*

Filter	*Short name*	*References*
Weighted averaging over dynamic time window	TWW	van der Kamp and Sundstedt [2011], Kumar et al. [2008]
Two-mode low-pass filter	LP2	Olsson [2007]
Savitzky-Golay	SG	Nyström and Holmqvist [2010]
Kalman	Kalman	Komogortsev and Khan [2007]
Weighted on-off filters	Woo	Jimenez et al. [2008], Veneri et al. [2010]
Alternative weighted on-off filter	AWoo	Chapter 2.1
W_g	W_g	Chapter 2.1

2.1 Modified filters

The original weighted on-off filter has calculation routine with signal mode detection (i.e., hold a buffer with points of an alternative fixation) that introduces a one-sample delay. In some case it may be unwanted, therefore an alternative (simplified) filter was tested as well. The simplification here means removing the alternative buffer and measuring only the distance between the previous and new output points.

Weighted Gaussian filter is another simplified version of the weighted on-off filter with kernel Gaussian function, which does not recognize fixation and saccade modes. The filter simply returns the weighted average of the last gaze points which have weights greater than 0.05.

2.2 Optimization

Wide range of parameters of each filter was tested in this study to find the best estimation values when applied to the collected data. Often, a change in a certain parameter improved one estimation parameter but worsened another. In such cases the selected parameters were not optimal if evaluated each separately.

3 Evaluation of filtering methods

Ideally, the filtered signal should consist of still fixations and fast saccades [Stampe 1993]. However, eyes slightly move during fixation and even the most precise and accurate eye trackers will not produce such idealized signal.

3.1 Idealized signal

The evaluation and comparison of filtering methods require shaping the given signal as close to the idealized one as possible. This task is not trivial, and next we describe the method used for "ideal" signal estimation. First, the initial signal should be split into ranges with relatively small changes inside each range, and notable changes of signal between extreme points of the adjacent ranges. Then the value of each point inside a range becomes the average of this point and $\pm N$ closest points. Taking N very large ($N \rightarrow \infty$) will result in substituting of all points in a range with its average value. We have tested the filters against two ideal-

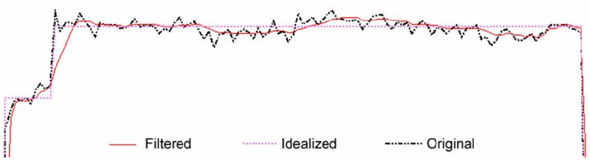

Figure 1 *Example of the original, idealized* (N_∞) *and filtered signals (X coordinate plotted against time).*

ized signals rather than one to decrease the impact of freely selected parameters on the final conclusions. The first idealized signal was constructed with $N = 7$, and the second with $N = \infty$.

Changes in signal between ranges occur about as quickly as in the original signal. Gaze points belonging to saccades may compose ranges of a single point. The split to ranges occurs in recursion: after the initial split is done, the algorithm calculates new data points, then a new split occur (based on new data points), and if new ranges differ from initial, another round of calculation takes place, until no changes detected between the previous and new sets of ranges.

3.2 Comparison criteria

The comparison of filters is based on three criteria: a) introduced delay D, b) smoothness S, and c) closeness to the idealized signal C. The delay D and closeness C are expressed as RMS of the distances between points of idealized and filtered signals located further (for D) or closer (for C) than some threshold. Smoothness S of the filtered signal is expressed as RMS of distances between actual and predicted data points of the filtered signal, if they are close. The predicted points are estimated by adding a difference between two previous points to the latest point.

The ideally smoothed signal is a line on a time-signal plot (see Figure 1), and its $S = 0$ (on the X-Y plot the gaze point moves with a constant speed and direction). Alternatively, smoothness can be estimated as the average distance between close adjacent points of the filtered signal (S_p). However, S_p is a stricter indicator of smoothness: its value is equal to 0 if the signal is not simply linear, but is also constant: $S_p = 0$ only when the gaze is still.

4 Data collection

Eleven volunteers from students or staff members at the local university took part in the test. All had experience in using eye-tracking technology. Tobii T60 eye tracking system (60Hz) was used to measure eye movements. The proprietary software with embedded ETU-Driver was developed (C#, .NET 2.0) to display targets and collect data. The chair was set so that the participant's eyes were at approximately 60 cm from the 17-inch monitor (screen resolution was 1280x1024 pixels).

The experimental software displayed grey full-screen window that was divided logically into 25 cells (5 by 5), and targets (6 by 6 pixels black squares) were presented in the center of each cell in random order. One block of the experiment consisted of presentation of 25 targets. Each participant completed two blocks of trials (i.e., 50 trials per participant).

The experiment was starting with eye tracker calibration and calibration verification. After the calibration was recognized as successful, the first block started by displaying a home box at the screen center for 2 seconds. The experiment continued displaying 25 targets one by one. Each target remained visible for 2 second; they were shown and hidden automatically. In order to escape visual search for targets appearing far from the previously shown target, a big white rectangle was displayed for 150 ms at the new target appearance location to make a "blink" effect. After a block was finished, participants had a chance to rest, if needed, and then press a key to continue with the next block. Gaze points (timestamp and location) and the displayed target properties (location and size) were logged into a file.

5 Results and Discussion

The relations between D, C, S and S_p for each filter are shown in Figure 2 when tested against the N_7 (left) and N_∞ (right) idealized signals. The size of a data point on this figure reflects the general smoothness S (outer lighted circle) and horizontal smoothness S_p (inner darker circle). Smaller values of each estimation parameter mean better evaluation of filtering capabilities. The estimation values for each filter are shown in Table 2.

Despite some differences between both graphs, the performance of most of the filters can be described unambiguously. Savytzki-Golay and Kalman filters have poor smoothness and closeness estimations comparing to others filters therefore these filters were recognized as the worst in this comparison. The weighted Gaussian filter (W_g) has high D and S with average C, and therefore can be ranked as relatively poor filter among the compared. The two-mode low-pass filter ($LP2$) has shown moderately good filtering: it has high C, but low-to-average D and S. Other stud-

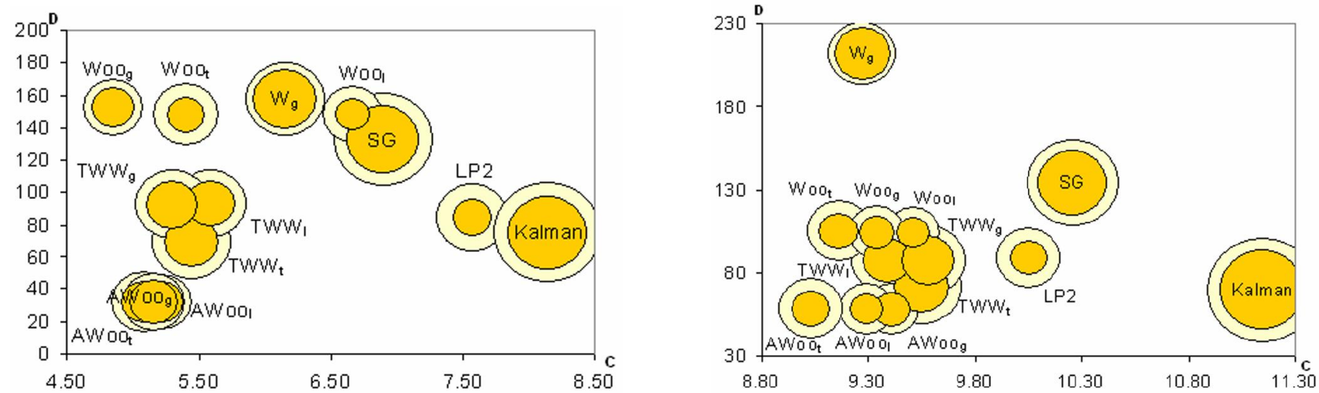

Figure 2 *Filters performance estimation when compared against* N_7 *(left) and* N_∞ *(right) idealized signal: Closeness (C) to the idealized signal, Delay (D), Smoothness (S and* S_p*, presented as diameters of outer and inner circles). Lower values indicate better evaluation.*

Table 2 *Estimation values of closeness C, delay D, smoothness S and alternative smoothness Sp (lower values are better)*

Filter	*C*	*D*	*S*	S_p
TWW_t	5.45	69.6	5.38	2.50
TWW_l	5.59	92.8	4.70	2.10
TWW_g	5.30	92.6	4.79	2.37
LP2	7.58	84.1	4.73	1.40
SG	6.90	132.6	8.36	4.69
Kalman	8.15	75.1	10.00	5.54
Woo_l	6.67	147.7	3.07	1.01
Woo_t	5.41	147.8	3.78	1.37
Woo_g	4.85	152.2	3.14	1.65
$AWoo_l$	5.22	32.14	3.17	1.79
$AWoo_t$	5.09	32.26	3.72	1.84
$AWoo_g$	5.16	32.14	3.38	2.01
W_g	6.16	157.9	5.41	3.51

ied filters showed typical trade-off between the estimation parameters: high *D*, low *S*, and low to average *C* for weighted on-off *Woo* filters, and low *C* and *D*, but average *S* for weighted average over dynamic time window filters (*TWW*). These filter families are clearly performing better than average, but no one can be recognized as significantly outperforming another.

All three estimation parameters calculated for the alternative weighted on-off filters (*AWoo*) are among the lowest. Its variant with triangular kernel function has shown the best relation between delay and closeness at a little expense of smoothness when tested against both idealized signals.

The result of comparison may lead to a conclusion that the key requirement for designing filters is detection of a signal state, while the other routine can be quite simple. Filters with signal state detection suppress smoothing when a saccade starts, therefore the introduced delay is minimal. If the averaging routine is based on kernel functions, then triangular or Gaussian function are recommended. However, a low-pass filter *LP2* is also can be considered due to its good smoothing performance.

6 Conclusions

Several filters of eye movement data found in literature were implemented and evaluated using delay, closeness to an idealized signal, and two smoothness criteria. Parameters of each filter were adjusted to ensure their best performance. The filters with signal mode detection (fixation/saccade) and based on weighted averaging over static (*[A]Woo*) or dynamically changing (*TWW*) time window were recognized as the best. However, no strict conclusions can be made about other filters except the clear outliers (Savytzki-Golay and Kalman fitters).

References

DUCHOWSKI, A, T. 2003 *Eye Tracking Methodology: Theory and Practice.* Springer-Verlag New York, Inc., Secaucus, NJ, USA.

GU, J., MENG, M., COOK, A., AND FAULKNER, M. G. 2000. Analysis of eye tracking movements using FIR median hybrid filters. In *Proceedings of the 2000 symposium on Eye tracking research & applications (ETRA '00)*, ACM, 65-69.

JACOB, R.J.K. 1993. Eye movement-based human-computer interaction techniques: Toward non-command interfaces. In *Hartson, H. R. and Hix, D. (Eds.) Advances in Human-Computer Interaction 4*, 151-190,

JIMENEZ, J., GUTIERREZ, D., AND LATORRE, P. 2008. Gaze-based Interaction for Virtual Environments. *Granitzer, M. et al. (Eds.), Journal of Universal Computer Science 14*, 19, 3085-3098.

VAN DER KAMP, J., & SUNDSTEDT, V. 2011. Gaze and Voice Controlled Drawing. In *Proceeding of The 1st conference on Novel Gaze-Controlled Applications (NGCA'11)*, ACM, #9.

KOHLER, M. 1997. *Using the Kalman Filter to track Human Interactive Motion-Modelling and Initialization of the Kalman Filter for Translational Motion*. Tech. Rep. #629, Informatik VII, University of Dortmund.

KOMOGORTSEV, O. V., & KHAN, J. I. 2007. Kalman filtering in the design of eye-gaze-guided computer interfaces. In *Proceedings of the 12th international conference on Human-computer interaction: intelligent multimodal interaction environments (HCI'07)*, Springer-Verlag, 679-689.

KUMAR, M., KLINGNER, J., PURANIK, R., WINOGRAD, T., AND PAEPCKE. A. 2008. Improving the accuracy of gaze input for interaction. In *Proceedings of the 2008 symposium on Eye tracking research & applications (ETRA '08)*, ACM, 65-68.

NYSTRÖM, M., & HOLMQVIST, K. 2010. An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data. *Behavior Research Methods 42*, 1, 188-204.

OLSSON, P. 2007. *Real-time and offline filters for eye tracking.* Master's thesis, Royal Institute of Technology, Sweden.

SAVITZKY, A., & GOLAY, M. 1964. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry 36*, 8, 1627-1639

SESIN, A., ADJOUADI, M., CABRERIZO, M., AYALA, M., AND BARRETO, A. 2008. Adaptive eye-gaze tracking using neural-network-based user profiles to assist people with motor disability. In *Yuhasz, S. C. (Ed.), Journal of Rehabilitation Research & Development 45*, 6, 801-818.

STAMPE, D. M. 1993. Heuristic filtering and reliable calibration methods for video-based pupil-tracking systems. *Behaviour Research Methods 25*, 2, 137-142

TOLE, J. R., AND YOUNG, L. R. 1981. Digital Filters for Saccade and Fixation Detection. In *Fisher, D. F. , Monty, R. A. and Senders, J. W. (Eds.) Eye Movements: Cognition and Visual Perception*, Lawrence Erlbaum Associates, Hillsdale, NJ, 7-17.

VENERI, G., FEDERIGHI, P., ROSINI, F., FEDERICO, A., AND RUFA, A. 2010. Influences of data filtering on human-computer interaction by gaze-contingent display and eye-tracking applications. *Computers in Human Behaviour 26*, 6, 1555-1563.

ZHANG, X., REN, X., AND ZHA. H. 2010. Modeling dwell-based eye pointing target acquisition. In *Proceedings of the 28th international conference on Human factors in computing systems (CHI '10)*, ACM, 2083-2092.