Pre-print of the article for the Journal of Eye Movement Research
1(1):1, 1-2

Quick Models for Saccade Amplitude Prediction

Oleg V. Komogortsev
Department of Computer Science
Texas State University-
San Marcos
ok11@txstate.edu

Young Sam Ryu
Ingram School of Engineering
Texas State University - San
Marcos
yryu@txstate.edu

Do H. Koh
Department of Computer
Science
Texas State University-
San Marcos
dk1132@txstate.edu

This paper presents two new saccade amplitude and direction prediction models. Both models were designed to predict saccade amplitude very quickly. One of the models predicts saccade amplitude by using the data from the first eye position sample at the beginning of a saccade's trajectory and the other model uses the data from the first two samples. Both models are based on a two state Kaman Filter. Models' prediction performance was tested with 35 subjects. The amplitude accuracy results yielded approximately 5.4° prediction error for both models, while the error for direction prediction was 5.3% for the first sample model and 1.5% for the two samples model. The practical use of the proposed models lays in the area of real-time gaze-contingent compression and extreme eye-gaze aware interaction applications.

Keywords: Saccade, prediction, Kalman filter, Human Computer Interaction

Introduction

Human Visual System (HVS) exhibits a variety of eye movements: fixations, saccades, smooth pursuit, optokinetic reflex, vestibulo-ocular reflex, and vergence [1]. When a person is sitting in front of the computer screen, usually only fixations, pursuits, and saccades are present. Among those three eye movements, saccades are the fastest movements transitioning the eye between relatively stable fixation spots [2]. The fixations provide the highest quality picture to the brain, while HVS is blind during saccades [2]. Pursuits are rarely exhibited when a person is working in front of a computer screen, i.e. pursuits appear when a person looks at objects with translational motion. Quality of vision during pursuit varies. Two areas inside of the Human Computer Interaction (HCI) domain - gaze contingent compression (GCC) systems [3-6] and systems with direct eye gaze control [7-10] - employ the characteristics of eye movements to make HCI systems more efficient and responsive.

Real-time GCC systems exploit the properties of the HVS where the area of the highest visual acuity is approximately 2° of the visual angle, while the quality of vision in the periphery is severely degraded [11]. Real-time GCC system tries to accurately estimate the location of the fixation spot estimated by an area called a Region of Interest (ROI). A challenge in front of any GCC is to minimize the ROI size, without letting a user to see artifacts introduced by the GCC compression. Network transmission of the multimedia content while GCC is performed induces various transmission delays into the system. As a result of the delay, saccades can place a gaze on the low quality coded part of the image [6]. Therefore, a real-time GCC system must have a saccade amplitude prediction algorithm that allows placing a high quality coded ROI on top of the future fixation/pursuit movement to compensate for the delay effects [6]. A quick amplitude prediction model can reduce the delay, therefore improving the performance of a real-time GCC system. Discussion section of the this paper provides a theoretical validation of this claim.

Today in the human computer interaction world, the mouse and the keyboard are the primary input devices. Recently eye-gaze aware interfaces, based on the eye tracker as an input device, have been gaining popularity in the HCI community [10, 12-15]. The majority of the HCI systems use fixation duration (dwell time) as a trigger for interface actions [8, 15, 16]. In such interfaces, the duration of the detected fixation triggers a “click”. Fixation-based selection necessitates data buffering, and therefore, introduces a delay in the system. Pursuit-based selection is an unexplored topic in the HCI community. Nevertheless, the definition of the pursuit implies that its detection will require a certain amount of data buffering. Due to their speed saccade selection would seem to be the most appropriate in the applications where the quickness of the target selection is of the utmost importance. Saccade’s characteristics would be employed to “click” a target even before the eye moves to the target’s location. Such a scheme would require that saccade’s amplitude and direction would be predicted based on the first few or even one eye position samples belonging to the saccade trajectory.

We are aware of only two previously published works that discussed a prediction of the saccade amplitude. The first work was authored by Anliker [17], where the author employed the fact that saccades in nature are ballistic [1], i.e., once the peak velocity is detected, the remaining saccade trajectory resembles the trajectory before the peak. The second work authored by Komogortsev and Khan (2007) employed a two state Kalman Filter (TSKF) for saccade amplitude prediction. This paper builds upon the work by Komogortsev and Khan (2007) and creates two new amplitude prediction models. The performance of the developed models is tested by 35 subjects with a stimuli designed to envoke saccades of various amplitudes.

Human Visual System Modeling by a Kalman filter

Kalman Filter has played an important role in the eye movement related research. Sauter et al. (1991) has proposed a mechanism for the detection of the saccade onset/offset based on the innovations generated by a Kalman Filter. Rewari and Chi-Sang (1993) have applied a general likelihood approach to improve detection for the saccades of small amplitudes Abd-almageed et al. (2002) has proposed parameters that allowed to more accurately reconstruct a pursuit signal in cases when the signal was corrupted by noise. Komogortsev and Khan (2007) have applied a Kalman filter both in a gaze contingent compression systems and systems with direct eye gaze input. In these systems Kalman Filter was employed as a predictor of visual attention and filter for eye position samples not detected by an eye tracker. Kalman Filter together with incorporated Oculomotor Plant Mechanical Model was employed as a predictor of the eye movement trajectories in cases when saccade amplitude was known [18, 19]. The specific focus of the current work is quickness of prediction and evaluation of the accuracy of such prediction.

Kalman Filter

The Kalman filter is a recursive estimator that computes a future estimate of the dynamic system state from a series of incomplete and noisy measurements. The Kalman Filter minimizes the error between the estimation of the system’s state and the actual system’s state. Only the estimated state from the previous time step and the new measurements are needed to compute the new state estimate. Many real dynamic systems do not exactly fit this model; however, because the Kalman filter is designed to operate in the presence of noise, an approximate fit is often adequate for the filter to be quite useful [20].

The Kalman Filter addresses the problem of trying to estimate the state $x \in \Re^n$ of a discrete-time controlled process that is governed by the linear stochastic difference equation [20]:

$$x_{k+1} = A_{k+1}x_k + B_{k+1}u_{k+1} + w_{k+1} \quad \textbf{(1)}$$

with the measurement

$$z_k = H_k x_k + v_k \quad \textbf{(2)}$$

The n-by-n state transition matrix A_{k+1} relates the state at the previous time step k to the state at the current step k+1 in the absence of either a driving function or process noise. B_{k+1} is an n-by-m control input matrix that relates m-by-l control vector u_{k+1} to the state x_k. w_k is an n-by-1 system's noise vector with an n-by-n covariance matrix Q_k. $p(w_k)\sim N(0,Q_k)$. The measurement vector z_k contains state variables that are measured by the instruments. H_k is a j-by-n observation model matrix which maps the state x_k into the measurement vector z_k. v_k is a measurement noise j-by-1 vector with covariance R_k. $p(v_k)\sim N(0,R_k)$.

While Equations (1)-(7) provide the mathematical description of the process that is being modeled the actual state values x_{k+1} are unknown and have to be estimated. The estimation of x_{k+1} requires two distinct phases Predict and Update [20].

Predict:

Predict the state vector ahead:

$$\hat{x}_{k+1}^{-} = A_{k+1}x_k + B_{k+1}u_{k+1} \qquad \textbf{(3)}$$

The $\hat{x}_{k+1}^{-}$ is a future estimation of the modeled state without a measurement from the measurement instrument. In case of the eye movement prediction the value of $\hat{x}_{k+1}^{-}$ can be employed as a predictor of the future gaze position.

One of the Kalman filter goals is to minimize the error between the actual state value x_{k+1} and the estimation of this value $\hat{x}_{k+1}$ (Equation (6)). For these purposes, the first estimate of the error covariance metrics is computed following mathematical representation of the modeled process:

$$P_{k+1}^{-} = A_{k+1}P_kA_{k+1}^{T} + Q_{k+1} \qquad \textbf{(4)}$$

Update:

The update phase improves the estimate of the modeled process by considering the measurement from the measurement device. The update state can be broken down into three distinct steps.

Compute the Kalman gain:

$$K_{k+1} = P_{k+1}^{-}H_{K+1}^{T}(H_{k+1}P_{k+1}^{-}H_{k+1}^{T} + R_{k+1})^{-1} \qquad \textbf{(5)}$$

Update the estimate of the state vector with a measurement z_{k+1}:

$$\hat{x}_{k+1} = \hat{x}_{k+1}^{-} + K_{k+1}(z_{k+1} - H_{k+1}\hat{x}_{k+1}^{-}) \qquad \textbf{(6)}$$

Update the error covariance matrix:

$$P_{k+1} = (I - K_{k+1}H_{k+1})P_{k+1}^{-} \qquad \textbf{(7)}$$

The choice of Kalman filter gain K_k allows to minimize the estimate error covariance $P_k = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)]$. A more detailed description of Kalman filter mechanics are beyond the scope of this paper and can be found in [20, 21].

Human Visual System

The approach that we use in this paper is to model an eye as a system with two states: position and velocity, with acceleration modeled as white noise with known maximum acceleration. Next we apply Kalman filter framework to this eye representation creating a Kalman filter with two states and calling it *Two State Kalman Filter* (TSKF). To complete the description of this filter we describe our choice for the state vector x_k, control vector u_k, transition matrix A_k, control matrix B_k. It is also necessary to derive a covariance matrix Q_k for the system's noise w_k and covariance matrix R_k defining the measurement noise v_k. Additionally, to map actual system's state vector x_k to the measurement vector z_k, observation matrix H_k is required. Following subsection provides detailed description below.

Note that 2D eye movement parameters (position, velocity, acceleration) can be broken into the vertical and horizontal components, because essentially they are composed of superposition of their respective orthogonal components [22]. Therefore, we create two instances of the TSKF filter: first is responsible for the horizontal component of movement and the second one is responsible for the vertical. As a result, an eye is represented as a system which has two state vectors x_k and y_k.

$$x_k = \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \qquad \textbf{(8)}$$

where $x_1(k)$ is horizontal coordinate of the gaze position and $x_2(k)$ is horizontal eye-velocity at time k.

$$y_k = \begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} \qquad \textbf{(9)}$$

where $y_1(k)$ is vertical gaze position and $y_2(k)$ is vertical eye-velocity at time k.

The state transition matrix for both horizontal and vertical states is:

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \qquad \textbf{(10)}$$

where Δt is the eye-tracker's eye-position sampling interval.

The observation model matrix for both state vectors is:

$$H = [1 \quad 0] \qquad \textbf{(11)}$$

By definition, the covariance matrix for the measurement noise is$R_k = E[(v_k - E(v_k))(v_k - E(v_k))^T]$. Because only the eye position is measured, v_k is a scalar making$R_k = VAR[v_k] = \delta_v^2$, where δ_v is the standard deviation of the measurement noise. In this paper, it is assumed that the standard deviation of the measurement noise relates to the accuracy of the eye tracker and is bounded by one degree of the visual angle. Therefore δ_v was conservatively set to1°. In cases when the eye tracker fails to detect eye position coordinates, the standard deviation of measurement noise is assigned to be $\delta_v = 120°$.The value of 120° is chosen empirically, allowing the Kalman Filter to rely more on the predicted eye position coordinate$\hat{x}_k^-$.

The TSKF is initialized with zero valued initial vectors$\hat{x}_0$, $\hat{y}_0$ and an identity error covariance matrix P_0.

By definition, the process noise covariance matrix is $Q_k = E[(w_k - E(w_k))(w_k - E(w_k))^T]$, where w_ka 1x2 system's noise vector is $w_k = [w_1(k) \quad w_2(k)]^T$. The TSKF Simple model assumes that variables $w_i(k)$ are uncorrelated between each other (velocity is independent of eye position), i.e., $E[(w_m(k)w_n(k)] = E[(w_m(k)]E[w_n(k)]$ for all $n \neq m$ and $p(w_1(k)) \sim N(0, \delta_1^2)$, $p(w_2(k)) \sim N(0, \delta_2^2)$. These assumptions generate the following system's noise covariance matrix: $Q_k = \begin{bmatrix} \delta_1^2 & 0 \\ 0 & \delta_2^2 \end{bmatrix}$. This simple model assumes that the standard deviation of the eye position noise $w_1(k)$ is connected to the characteristics of the eye-fixation movement. Each eye fixation consists of three basic eye-sub-movements: drifts, small involuntary saccades and tremors [23]. Among those three movements, involuntary saccades have the highest amplitude—about a half degree of the visual angle; therefore, δ_1 is set conservatively to 1°. The standard deviation value for eye velocity was selected to be $\delta_2 = 1°/\mathrm{s}$.

Chi-square Test & Saccade Amplitude

The Chi-square test was originally employed by Sauter [24] to detect the onset and the offset of a saccade. The Chi-square test monitors the difference between predicted and observed eye velocity:

$$\chi_i^2 = \sum_{i=1}^{p} \frac{\left(\hat{x}_2^-(i) - \dot{\theta}_i\right)^2}{\delta^2} \qquad (12)$$

where $\hat{x}_2^-(i)$ is the predicted eye velocity computed by Equation (3) and $\dot{\theta}_i$ is the observed eye velocity. It is important to note that Equation (3) can be computed as a result of the Kalman filter framework presented by Equations (1)-(7) and specifically defined by equations (8)-(11). δ is the standard deviation of the measured eye velocity during the sampling interval under consideration. Once a certain threshold of the χ_i^2 is achieved (value of 25 is used in our system), a saccade is detected. It was reported that the filter stability improves if δ is selected to be a constant [25]. Empirical evaluation has indicated that values of $\delta^2 = 1000$ and p=5 provide acceptable performance.

Komogortsev and Khan (2007) have suggested a function that connected the value of χ_i^2 to the amplitude of the corresponding saccade (2007). They suggested that the development of such a function is possible due to the fact that HVS uses phasic (fast) eye-muscle fibers with high motoneuronal firing rate for large saccades and tonic (slow) eye-muscle fibers with a lower motoneuronal firing rate for the saccades of lesser amplitude [26]. Such mechanism ensures different rate of rise of eye-muscle force for the saccades of various amplitudes providing higher acceleration to the eye globe during saccades of higher amplitude. Larger amplitudes produce larger eye velocity values therefore increasing the value for χ_i^2.

This paper presents more robust models that derive a saccade's amplitude from a χ_i^2 value than the model originally proposed by [25]. The detailed description of those prediction models is provided in the section below.

Saccade amplitude Prediction

Velocity Model

As a base comparison model, we would like to employ a saccade prediction model proposed by Anliker [17]. Anliker's model uses the fact that saccadic movement is ballistic—i.e., a saccade trajectory is predetermined and cannot be altered once the movement starts, and the saccade trajectory resembles a bell curve. Once the peak velocity is reached, the rest of the saccade movement mirrors the movement prior to the peak. In our implementation of the Anliker's model, the velocity peak is detected when a consecutive eye position point has a lesser absolute velocity value than the previous absolute velocity value. The saccade amplitude is made equal to double the distance the path traveled, prior to the velocity peak.

First Sample Model

The Chi-square test was performed by Equation (12). When a χ_i^2 went above a threshold an onset of a saccade was recorded and the resulting saccade amplitude was stored for the analysis. Subsequently, the saccade amplitude prediction function was derived by a non-linear regression model with Gauss-Newton method implemented in a SAS system. Nonlinear regression is a powerful tool for analyzing scientific data, especially in physiology [27]. Also, nonlinear regression is more effective for curve-fitting than linear regression. This method provided the equation that connected the Chi-square test value to the predicted saccade's amplitude, A_{sac_pre}:

$$\begin{aligned} A_{sac\_pre} = {} & -119 \cdot 10^{-14} \cdot (\chi^2)^5 + 2.166 \\ & \cdot 10^{-9} \cdot (\chi^2)^4 - 1.3 \\ & \cdot 10^{-6} \cdot (\chi^2)^3 \\ & + 0.000282 \cdot (\chi^2)^2 \\ & - 0.0117 \cdot (\chi^2) \\ & + 10.5328 \end{aligned} \tag{13}$$

The terms with power of more than 5 were dropped, due to non-significant impact contributing to the final value.

Two Samples Model

Two Samples model employed first two Chi-square test values for estimation. The first one (χ_1) was recorded immediately at the saccade onset point, the second one (χ_2) was recorded for the next consecutive eye position sample. The non-linear regression model with the Gauss-Newton method was used to derive the following estimation of the saccade amplitude, A_{sac_pre}.

$$\begin{aligned} Sac_{amp} = {} & -107 \cdot 10^{-13} \cdot ({\chi_1}^2)^5 + 1.848 \\ & \cdot 10^{-8} \cdot ({\chi_1}^2)^4 \\ & - 0.00001 \cdot ({\chi_1}^2)^3 \\ & + 0.000266 \cdot ({\chi_1}^2)^2 \\ & - 0.3008 \cdot ({\chi_1}^2) - 179 \\ & \cdot 10^{-15} \cdot ({\chi_2}^2)^5 + 5.36 \\ & \cdot 10^{-10} \cdot ({\chi_2}^2)^4 - 4.8 \\ & \cdot 10^{-7} \cdot ({\chi_2}^2)^3 \\ & + 0.000084 \cdot ({\chi_2}^2)^2 \\ & + 0.0586 \cdot ({\chi_2}^2) \\ & + 9.1502 \end{aligned} \tag{14}$$

Saccade Direction Detection

This paper tested only saccades with horizontal amplitude. Quick direction detection of this movement was not as easy as it might appear due to equipment noise. Quick direction detection schemes are prone to generate errors. In the next two paragraphs, we present two methods for the horizontal saccade's direction detection

First Sample Model

The direction of movement was connected to the sign of the velocity of the recorded signal. Positive sign of the velocity signal at the first eye position sample indicated a rightward direction of a saccade, and the negative velocity indicated a leftward direction of the saccade.

Two Samples Model

Two first velocity samples of the saccade trajectory were evaluated. The rightward saccade was predicted if both velocity samples had positive values. The leftward saccade was predicted if both velocity samples had negative values. In case when velocity samples had different signs, the saccade direction was selected to be the same as the sign of the velocity point with the highest absolute value.

Methodology

Equipment

The experiments were conducted with the Tobii x120 eye tracker, which is represented by a standalone unit connected to a 19 inch flat panel with a resolution of 1280x1024. This eye tracker performs binocular tracking with the following characteristics: accuracy 0.5°, spatial resolution 0.2°, and drift 0.3°. Tobii x120 model allows 300x220x300 mm freedom of the head movement. Nevertheless a chin rest was employed for higher accuracy.

Stimulus Presentation

A saccade inhibition stimulus was presented as a ramp stimulus [1] where a dot appeared at the random horizontal location on the screen (vertical coordinate was fixed to the center of the screen). First, the dot flashed for 1000 ms.; then it disappeared; then, immediately a new dot appeared at the new, random location. The minimum distance between two consecutive dots was 2°. Each subject was presented with a sequence of 30 dots.

Participants

Thirty five college students were recruited in undergraduate courses at Texas State University. Participants were compensated for their participation with extra credit in courses within the Departments of Psychology and Computer Science. All materials and procedures were approved by the Institutional Review Board at Texas State University, and informed consent was obtained from all participants prior to the testing session. On average, participants were 20.62 years of age [SD = 2 years; range = 18-25]. Of the 38 participants tested, 85% were male, 69% were of European-American descent.

Quality of the Recorded Data

Prior to the experiment, participants were screened for the actual accuracy and noise levels of the eye-tracker hardware using software developed in the Human Computer Interaction Laboratory at Texas State University [28]. Participants with reported accuracy of less than 1° and a noise level of more than 16% were excluded from the analysis of eye movement data. The noise level is defined as the percentage of eye position samples for which the eye tracker failed to report the eye position coordinates. Some recording failures of the eye tracker occurred, due to conditions such as squinting and excessive moisture of the eye. In eye-tracking experiments, noise level parameters are rarely reported, but it serves as a major validation metric that should be specifically stated to verify the validity of the results.

Evaluation Metrics

The *Root Mean Squared Error* (RMSE) between the predicted A_{sac_pre} and the actual saccade amplitude A_{sac} determines the accuracy of the saccade prediction algorithm. $RMSE^M = \sqrt{\sum_{k=1}^{N} \frac{(A_{sac} - A_{sac\_pre})^2}{N}}$. M is the model's name. The ideal saccade prediction model will have the RMSE of 0º.

Direction Prediction Error (DPE) represents the amount of erroneously detected saccade prediction—i.e., the rightward saccade was predicted as leftward saccade and otherwise. The perfect scheme would have an error rate of 0.

Both metrics were computed in the following way. Out of 35 recordings, 25 were randomly selected to create functions connecting the Chi-square test value to the saccade amplitude according to heuristic of each prediction model. The remaining 10 recordings were employed to compute the RMSE and DPE metrics.

Results

Amplitude Prediction Error

Velocity Model yielded average RMSE of 3.46°. We hypothesize that these errors were due to high noise in the eye tracker. For example, spatial resolution of 0.2° results in the velocity noise of 24°/s without an eye effectively moving anywhere. Average RMSE for the First Sample Model was 5.41°. The second Chi-square test sample did not improve the accuracy of amplitude prediction significantly from the First Sample Model– average RMSE for the Two Samples Model was 5.45°. However, RMSE for Velocity Model was significantly lower than those for the other two models, $F(2,18)=15.20$, $p<0.01$.

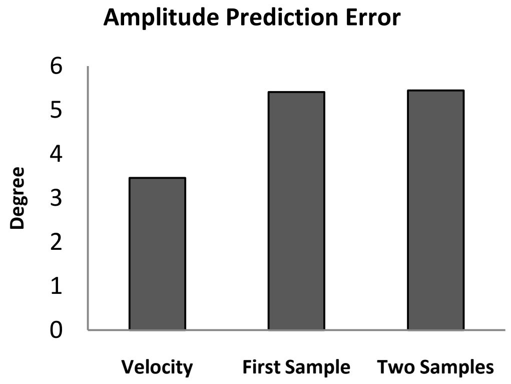

Figure 1. Saccade amplitude prediction errors for various prediction models by average RMSE

Direction Prediction Error

Average error rate for the First Sample Model for all 35 recordings was 5.26%. Two Samples Model performed significantly better by reducing average error rate to 1.54%, *F*(1,34)=25.87, *p*<0.01.

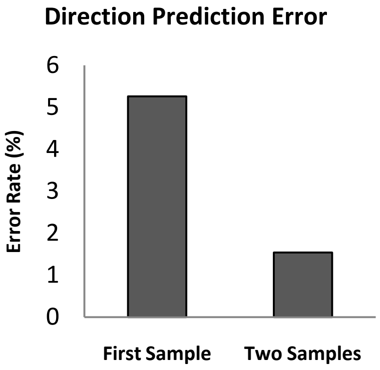

Figure 2. Direction prediction error rates for First Sample and Two Samples models.

Discussion

Prediction Accuracy Challenges

Average error of the best performing model with the amplitude of 5.41° seems to be large for the average saccade amplitude of 10°. The explanation of this fact may be presented by the graph depicted in Figure 3.

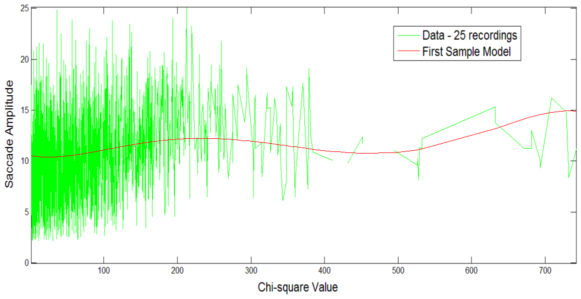

Figure 3. Recorded saccades' amplitudes for computed Chi-squared values for 25 randomly selected records

The graph indicates that the First Sample model's equation splits the recorded data in approximately two halves. Thus, an additional research is required to find data noise reduction algorithms that will allow reducing the noise in the recorded data and increasing the accuracy of prediction.

In previous research, Komogortsev and Khan proposed following equation for saccade amplitude prediction

$$A_{sac\_pre} = -24 \cdot 10^{-6} \cdot (\chi^2)^3 + 536 \cdot 10^{-4} \cdot (\chi^2)^2 + 1.5 \qquad \textbf{(15)}$$

when this equation was applied to the data collected in the experiments described in this paper, the recorded average RMSE was more than 35° (larger than the monitor size). One possible reason for such low accuracy of prediction was due to the different frequency of eye position sampling recording in our experiments and in experiments presented in Komogortsev and Khan (2007). The frequency employed in their experiment was 50Hz and frequency employed in our experiments was 120Hz. The other possible issue is that the function presented in the Equation (15) was derived empirically only from one recording of one subject.

Applications

Quick models for saccade amplitude prediction have potential to benefit the area of gaze-contingent-compression, by reducing the amount of lag created by sensing, processing, and transmission delays. As it was pointed out in our previous work [29, 30] delay reduction allows to achieve higher levels of compression. Specifically an amplitude prediction model would provide the coordinates for the high quality ROI region placed on the fixation spot at the end of the saccade. The quickness of the saccade's amplitude prediction would result in delay reduction. Below we provide a mathematical evaluation of the lag reduction provided by every model.

Saccade duration can be estimated by the formula

$$D_{sac\_dur} = (2.2D_{sa\,c_{amp}} + 21)/1000 \qquad \textbf{(16)}$$

were D_{sac_amp} is an amplitude of a saccade measured in the degrees of the visual angle and D_{sac_dur} is a saccade duration measured in seconds. It is possible to estimate the amount of lag reduction each amplitude prediction model provides. **Velocity Model:** requires peak velocity identification that occurs at the point representing the middle of the saccade plus and additional point required to verify the peak, therefore prediction time is: $D_{sac\_dur}/2 + 1/f$ where f is the frequency of the eye tracker equipment measured in Hz. Total lag reduction is computed as:

$$T_{delay\_reduc} = D_{sac\_dur} - (D_{sac\_dur}/2 + 1/f)$$

One Sample Model: requires only one position at the beginning of the saccade, taking $1/f$ amount of time for prediction. Therefore, lag reduction can be computed as

$$T_{delay\_reduc} = D_{sac\_dur} - 1/f$$

Two Sample Model: requires two eye position samples, taking $2/f$ amount of time for prediction. Therefore, lag reduction can be computed as

$$T_{delay\_reduc} = D_{sac\_dur} - 2/f$$

Figure 4 provides the delay reduction performance depending of the saccades amplitude.

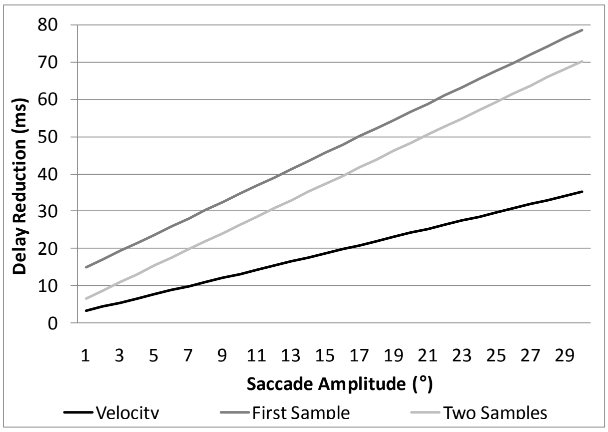

Figure 4. Delay reduction based on saccade's amplitude.

Once delay reduction is estimated, it is possible to assess amount of compression savings the models would provide for a GCC system. To provide such an estimate we have taken GCC performance results reported by Komogortsev [30]. Table 1 provides result summary

Delay	*0 ms*	*20 ms*	*500 ms*	*1000 ms*	*2000 ms*
APRG	2.6	2.2	1.6	1.5	1.4

Table 1. Average Perceptual Resolution Gain achieved with corresponding delay.

Average Perceptual Resolution Gain (APRG) is a quantity that indicates the amount of compression gaze-contingent approach provides given the value of the delay [29]. It is possible to provide a logarithmic approximation of APRG given the delay values

$$APRG_{orig} = -0.176\ln(T_{delay}) + 1.5 \qquad \textbf{(17)}$$

T_{delay} represents the amount of delay in GCC system measured in seconds. While Equation (17) provides the APRG estimation for the "Original" (no delay compensation) type of the GCC system, the GCC system's performance aided by amplitude prediction model can be presented as:

$$APRG_{mod} = -0.176\ln(T_{delay} - T_{delay\_reduc}) + 1.5 \qquad \textbf{(18)}$$

Resulting compression increase (CI) can be estimated as

$$CI = 100(1 - \frac{APRG_{orig}}{APRG_{mod}}) \qquad \textbf{(19)}$$

Figure 5 presents comparison results created by two scenarios defined by the average amplitude of the exhibited saccades. 1st is a 5° case and the 2nd scenario is a 10° case.

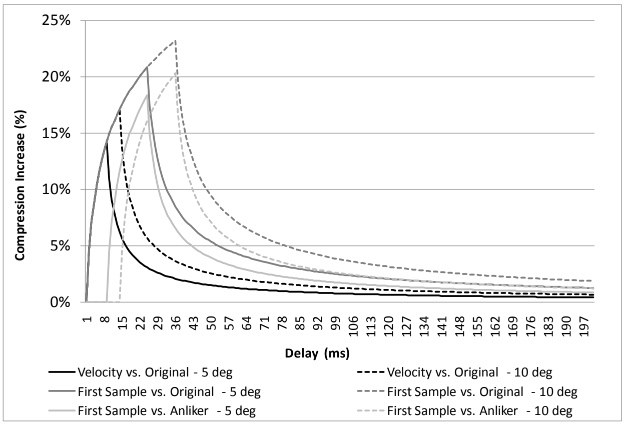

Figure 5. Compression increase as result of delay compensation.

Average saccade amplitude of 5°: the results indicate that Velocity model would provide a 14% compression improvement for the delay of 9 ms. There is a steep reduction in performance after this value, e.g., delay of 50 ms. achieves an improvement of performance of just 1%. First sample model increases the effective range of delay compensation, providing the highest increase of performance for the delays of up to 26 ms., there is a steep reduction in performance after that e.g., delay of 50 ms. reduces CI value down to 5% and delay of 100 ms. reduces the performance increase down to 2%. When performance of the First Sample model is compared to the performance of the Velocity model, the performance of both models is the same up to 8 ms. mark (the timing of the first interval) after this point we can see a constant rise of the performance until the point of 23-25ms reaching the value of 18%, there is a slow reduction of performance after this point.

Considering average saccade amplitude of 10° the performance of the Velocity and One Sample models are the same up to 14 ms. after this Velocity model performance degrades going down to 3% at the level of 50 ms., the performance reaches the mark of 1% or less at the point of 85ms. The peak of performance improvement occurs at 33-36 ms interval for the First Sample model, going down to the mark of 9% at the 50 ms. delay interval and 4% at 100 ms delay interval. When First Sample model is compared with Velocity model, the peak of performance improvement occurs at the interval of 34-36 ms providing an improvement of compression by 20%, the performance improvement decreases after this value.

Delay value indicate that Velocity model would provide a 14% compression improvement for the delay of 9 ms. There is a steep reduction in performance after this value, e.g., delay of 50 ms. achieves an improvement of performance by 1%. First sample model increases the effective range of delay compensation, providing the highest increase of performance of delay of 26 ms., there is a steep reduction of performance after this value e.g., delay of 50 ms. achieves an improvement of performance by 5%, delay of 100 ms. reduces the performance increase down to 2%. When performance of the First Sample model is compared to the performance of the Velocity model, the performance of both models is the same up to 8ms mark (the timing of the first interval) after this point we can see a constant rise of the performance until the point of 23-25ms reaching the value of 18%, there is a slow reduction of performance after this point.

We have provided a theoretical evaluation of the impact that quick saccade amplitude prediction models would have on the gaze-contingent compression domain. The actual implementation of such models in a real-time GCC system might change the estimated compression results. For example such factors as delay jitter, targeted amount of eye-gaze samples required to be contained by a GCC system [30] and prediction error compensation mechanisms were not considered in our evaluation. The detailed empirical evaluation of these factors is beyond the scope of this paper.

Conclusion

This paper has introduced two new saccade amplitude prediction models and compared them to the published work of Anliker [17]. New models were based on the Chi-square test values determined by a two state Kalman filter implementation of the Human Visual System. The results presented in this paper indicate that the Anliker's model was the most accurate producing a 3.46 degree error on average, but the model required that the middle of a saccade would be reached, therefore reducing the potential benefit of saving time. The Kalman filter-based models have produced a higher saccade amplitude prediction error. The model that required just one eye position sample at the beginning of a saccade for prediction produced an average error of 5.41° with saccade direction erroneously predicted 5% of the time. This was the fastest model in terms of amount of samples required for prediction. The second model employed two position samples, yielding the average amplitude prediction error of 5.45°, with just 1% error in saccade direction prediction. The exact application of the proposed saccade amplitude prediction models is beyond the scope of this paper. In each specific case, a designer of a gaze contingent compression system or a system with direct eye gaze control should decide which saccade prediction model is more appropriate.

REFERENCES

[1] R. J. Leigh and D. S. Zee, *The Neurology of Eye Movements*: Oxford University Press, 2006.

[2] A. Duchowski, *Eye Tracking Methodology: Theory and Practice* 2nd ed.: Springer, 2007.

[3] D. Parkhurst, J. and E. Niebur, "Variable resolution displays: A theoretical, practical, and behavioral evaluation," *Human Factors,* vol. 44, pp. 611-629, 2002.

[4] E. Reingold, M., L. Loschky, C., G. McConkie, W., and D. Stampe, M., "Gaze-Contingent multi-resolutional displays: An integrative review," *Human Factors,* vol. 45, pp. 307-328, 2003.

[5] A. Duchowski, T. and A. Çöltekin, "Foveated gaze-contingent displays for peripheral LOD management, 3D visualization, and stereo imaging.," *ACM Transactions on Multimedia Computing, Communications, and Applications,* vol. 3, 2007.

[6] O. V. Komogortsev and J. I. Khan, "Predictive real-time perceptual compression based on eye-gaze-position analysis," *ACM Trans. Multimedia Comput. Commun. Appl.,* vol. 4, pp. 1-16, 2008.

[7] R. J. K. Jacob, "What you look at is what you get: eye movement-based interaction techniques," in *Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people* Seattle, Washington, United States: ACM, 1990.

[8] S. Zhai, C. Morimoto, and S. Ihde, "Manual and gaze input cascaded (MAGIC) pointing," in *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit* Pittsburgh, Pennsylvania, United States: ACM, 1999.

[9] M. Kumar and T. Winograd, "Gaze-enhanced scrolling techniques," in *Proceedings of the 20th annual ACM symposium on User interface software and technology* Newport, Rhode Island, USA: ACM, 2007.

[10] A. Huckauf and M. H. Urbina, "On object selection in gaze controlled environments," *Journal of Eye Movement Research,* vol. 2, pp. 1-7, 2008.

[11] D. E. Irwin, "Visual Memory Within and Across Fixations," *Eye movements and Visual Cognition: Scene Preparation And Reading,* pp. 146-165, 1992.

[12] A. Huckauf and M. H. Urbina, "Gazing with pEYEs: towards a universal input for various applications," in *Proceedings of the 2008 symposium on Eye tracking research & applications* Savannah, Georgia: ACM, 2008.

[13] I. S. MacKenzie and X. Zhang, "Eye typing using word and letter prediction and a fixation algorithm," in *Proceedings of the 2008 symposium on Eye tracking research & applications* Savannah, Georgia: ACM, 2008.

[14] H. Istance, R. Bates, A. Hyrskykari, and S. Vickers, "Snap clutch, a moded approach to solving the Midas touch problem," in *Proceedings of the 2008 symposium on Eye tracking research \&\#38; applications* Savannah, Georgia: ACM, 2008.

[15] M. Kumar, A. Paepcke, and T. Winograd, "EyePoint: practical pointing and selection using gaze and keyboard," in *Proceedings of the SIGCHI conference on Human factors in computing systems* San Jose, California, USA: ACM, 2007.

[16] L. E. Sibert and R. J. K. Jacob, "Evaluation of eye gaze interaction," in *Proceedings of the SIGCHI conference on Human factors in computing systems* The Hague, The Netherlands: ACM, 2000.

[17] J. Anliker, "Eye Movements: On-Line Measurement, Analysis, and Control," in *Eye Movements and Psychological Processes,* R. A. Monty and J. W. Senders, Eds. Hillsdale: Lawrence Erlbaum Associates,, 1976, pp. 185–202.

[18] O. Komogortsev, V. and J. Khan, "Eye Movement Prediction by Kalman Filter with Integrated Linear Horizontal Oculomotor Plant Mechanical Model," in *Eye Tracking Research & Applications Symposium 2008,* Savannah, GA, 2008, pp. 229-236.

[19] O. V. Komogortsev and J. Khan, "Eye Movement Prediction by Oculomotor Plant Kalman Filter with Brainstem Control," *Journal of Control Theory and Applications,* vol. 7, pp. 14-22, 2009.

[20] R. Brown and P. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*. New York: John Wiley and Sons, 1997.

[21] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME вЂ" Journal of Basic Engineering,* pp. 35-45, 1960.

[22] M. Kohler, "Using the Kalman Filter to track human interactive motion --- Modelling and initialization of the Kalman Filter for translational motion," 1997.

[23] L. Yarbus, *Eye Movements and Vision*. Moscow: Institute for Problems of Information Transmission Academy of Sciences of the USSR, 1967.

[24] D. Sauter, M. B. J., N. Di Renzo, and C. Vomscheid, "Analysis of eye tracking movements using innovations generated by a Kalman filter," *Med. Biol. Eng. Comput.*, pp. 63–69, 1991.

[25] O. V. Komogortsev and J. Khan, "Kalman Filtering in the Design of Eye-Gaze-Guided Computer Interfaces," in *12th International Conference on Human-Computer Interaction (HCI 2007),* Beijing, China, 2007, pp. 1-10.

[26] A. T. Bahill, "Development, validation and sensitivity analyses of human eye movement models," *CRC Critical Reviews in Bioengineering,* vol. 4, pp. 311-355, 1980.

[27] H. Motulsky, "The GraphPad Guide to Nonlinear Regression," GraphPad Software, 2001.

[28] O. V. Komogortsev and S. M. Gowda, "Eye Tracker Calibration Accuracy Evaluation Tool

". vol. 2009, 2008.

[29] O. V. Komogortsev and J. Khan, "Predictive Perceptual Compression for Real Time Video Communication," in *12th ACM International conference on Multimedia (ACM MM 04),* New York, 2004, pp. 220-227.

[30] O. V. Komogortsev, "Gaze-Contingent Video Compression with Targeted Gaze Containment Performance," *Journal of Electronic Imaging,* in press.