User Target Intention Recognition from Cursor Position Using Kalman Filter

Gökçen Aslan Aydemir[1], Patrick M. Langdon[1], and Simon Godsill[2]

[1] Engineering Design Centre, University of Cambridge, Cambridge, United Kingdom
gkaslan@gmail.com, pml24@cam.ac.uk
[2] Engineering Department, Cambridge, United Kingdom
sjg@eng.cam.ac.uk

Abstract. This paper discusses user target intention recognition algorithms for pointing – clicking tasks to reduce users' pointing time and difficulty. Predicting targets by comparing the bearing angles to targets proposed as one of the first algorithms [1] is compared with a Kalman Filter prediction algorithm. Accuracy and sensitivity of prediction are used as performance criteria. The outcomes of a standard point and click experiment are used for performance comparison, collected from both able-bodied and impaired users.

Keywords: Intention Recognition, cursor movement, tracking, Kalman Filter.

1 Introduction

Human - computer interaction has become an every-day aspect of lives as technological devices make their ways out of the laboratory. New input devices are being introduced to the market not only in the form of computers and smart phones, but televisions, kiosks in hospitals, even advertisement screens in airports. The availability of these devices to ordinary people increased the diversity of the population using them, which include impaired and elderly users as well as able bodied, expert and non-expert users. Most of the input devices require a pointing task which can be difficult to some users, especially users with motor impairments which can become an overwhelming activity that they want to avoid.

Characteristics of cursor movement have been examined by HCI researchers in the recent years [2]. Algorithms to reduce the movement time and distance to target were suggested considering cursor movement characteristics of able-bodied users. Fitts' Law has been used as the basis in determining difficulty of a task depending on the target size and distance. It has been shown that it is possible to reduce this difficulty by increasing the target size [3-4], employing larger cursor activation areas, moving targets closer to cursor location, dragging cursor to nearest target or changing CD ratio [5]. However interactive systems employ more than several targets at a time and they are becoming more complex everyday. Hence, it is not trivial to decide on which target to expand or drag the cursor to. In addition,

C. Stephanidis and M. Antona (Eds.): UAHCI/HCII 2013, Part I, LNCS 8009, pp. 419–426, 2013.

pointing tasks could become more overwhelming for users if wrong targets are altered one after another. In the vicinity of a target prediction algorithm, dynamical alterations will become more meaningful and successful. Several algorithms have been proposed so far for target prediction.

However, cursor movement vary in characteristics for motor impaired users since they experience tremor, muscular spasms and weakness [8]. The velocity profile includes several stops and jerky movements. This needs to be taken into account when applying target prediction. State space filtering techniques are promising [9-9] in estimating intended targets as well as smoothing cursor trajectories since it is possible to model the movement, fine-tune or adapt the parameters for different users.

In this paper we compare a Kalman Filter framework with another algorithm suggested by Murata [1]. The accuracy and sensitivity of Murata's algorithm and a basic Kalman Filter framework will be investigated and compared using point-click task data from able-bodied and impaired users.

2 Method

Cursor data was collected from able bodied and motor impaired users, utilizing a standard point and click ISO task. Collection of start and end points were available for over 400 tasks. The user starts the task by clicking on a button at the centre of the screen and ends it by clicking at a target appearing on the screen. To evaluate the performance of intention recognition, 12 targets are located artificially around the starting point, comprising a circle and the end point being one of the targets. Intended target is indicated as target number 1 throughout this paper. The algorithms were run offline for comparison.

3 Target Intention Recognition from Bearing Angle

One of the first algorithms to be suggested was by Murata. The angle deviations towards all possible targets are calculated and the target with minimum deviation is determined. The results show that the pointing time is reduced by 25%. [1]. Asano et.al. point out that having more than one target on a particular movement direction results in poor performances of the afore mentioned algorithm, especially when dealing with target located far away. They used previous research results about kinematics of pointing tasks and showed that peak velocity and target distance have a linear relationship. They predict the endpoint with linear regression using the relationship between peak velocity and total distance to endpoint [6]. Lank et.al also employ motion kinematics where they assume minimum jerk law for pointing motion and fit a quadratic function to partial trajectory to predict endpoint [7].

However, Murata's method can provide a suitable starting point for target prediction since linear the motion assumption holds for especially able-bodied users and it is still one of the few and reasonably effective algorithms.

The idea is based on the assumption that a cursor should travel the shortest path, which is a line from the starting point to the intended target. This means that the angle from the current position to a possible end point should stay constant. The algorithm looks for a minimum change in angle calculated cumulatively.

The angle between two vectors, first being current cursor position – previous cursor position and second being current cursor position – target location is calculated (Fig.1). Target with minimum cumulative angle is determined as the prediction. This cumulative approach is important for impaired user cursor traces since cursor characteristics include a lot of deviations and a history should be available. It is also possible to apply weighing to buffer elements as discussed in Wobbrock's angle based CD Ratio adjustment methodology[5].

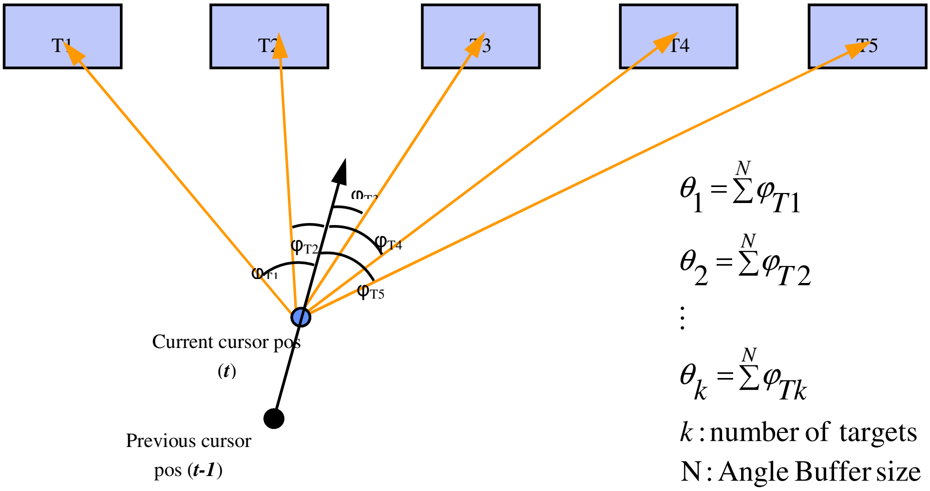

Fig. 1. Angles to targets are used to predict user's intention in Murata's algorithm

Figure 2 shows a cursor trace from a task completed by an able-bodied user. Correct prediction was made at the third data point since the trace is a straight line from the origin towards the target.

However, for more extreme cases such as overshooting targets and more complicated layouts with targets along the same direction, the performance is reduced. Comparative results are presented in the conclusion section. It is possible to increase the buffer size for disabled users to keep a longer history if the movement and increase accuracy at the cost of memory and computational complexity.

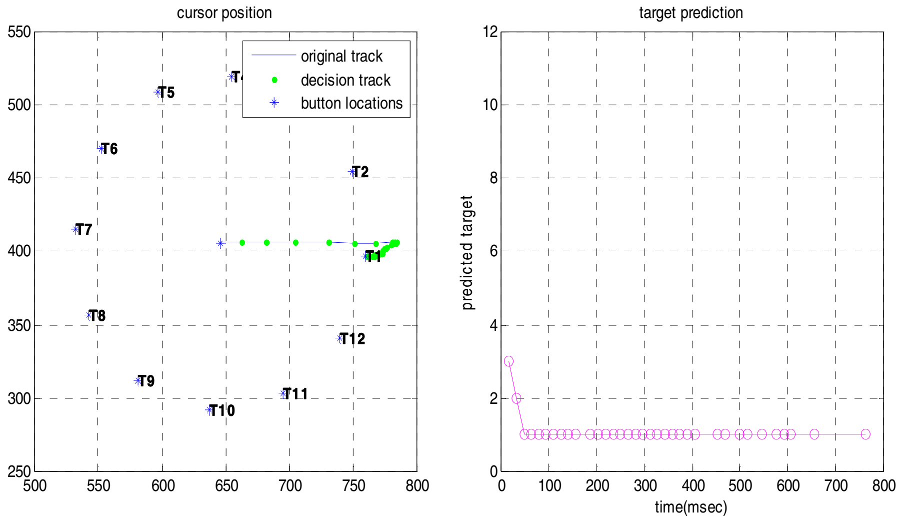

Fig. 2. Cursor coordinates collected from an able-bodied user and target prediction obtained by Murata's algorithm

4 Target Intention Recognition with Kalman Filter

Cursor trajectories of impaired motion have been shown to consist of sub-movements where the user stops and starts moving again with a different velocity and heading angle [10]. If this motion can be modeled in state-space, a Bayesian framework, such as a Kalman Filter, can be used for state space estimation, which can also work as a smoothing filter to aid the user visually [11]. In this paper, a basic Kalman Filter is proposed to estimate the cursor position and use the estimates to update a probability distribution for possible targets.

Looking at the velocity profiles of motor-impaired cursor trajectories, it could be a reasonable starting point to model the process as a "*nearly constant velocity*" process. The velocity is modeled as a Brownian motion and process equations are given in Equation (1) below.

Process model:

$$x[k+1] = x[k] + \Delta t * v_x[k] + \upsilon_x[k]$$

$$y[k+1] = y[k] + \Delta t * v_y[k] + \upsilon_y[k]$$

$$v_x[k] = dB$$

$$v_y[k] = dB \qquad (1)$$

Cursor position in horizontal and vertical directions is taken as measurements with white noise (Eqn.2). The noise characteristics can be personalized depending on the motion-impairment level of the user to account for jittery movements.

$$\mathbf{z}[k] = (x[k]\ y[k])' + \omega[k] \qquad (2)$$

It is possible to assume that all targets are equally probable at the beginning of any pointing task and update the probabilities. For N targets, the probability of target *i* will be:

$$p_i = 1/N \tag{3}$$

These probabilities can be updated according to the angle, distance and both angle and distance to targets at every measurement as given in Equation (4).

$$p_i[k+1] = p_i[k]*(1/\text{distance\_to\_target})$$

$$p_i[k+1] = p_i[k]*(1/\text{angle\_to\_target})$$

$$p_i[k+1] = p_i[k]*(1/\text{distance\_to\_target})*(1/\text{angle\_to\_target}) \tag{4}$$

For the following sections KA will be used to indicate prediction using only angle to target, KD to indicate prediction using only distance to target and KAD to indicate prediction using them together.

Experiment results show that using both distance and angle to target to update the target probabilities can handle different cases more robustly. Figure 3 shows the cursor coordinate estimates and measurements on the left and the target predictions from a task using Kalman Filter. It can be seen from the graph that KA and KD can result in wrong predictions more often whereas using KAD provides the correct prediction at a reasonable time and distance. Figure 4 shows the cursor trace in 3d, z-axis being the time. The instance of first correct prediction from KAD is marked with a data tip.

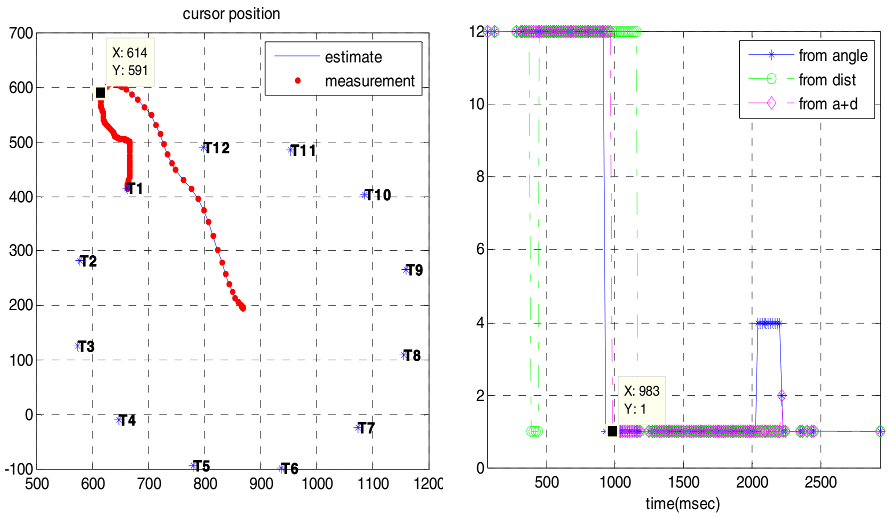

Fig. 3. Cursor trace and target predictions. Using angle and distance to target together produces a more robust result.

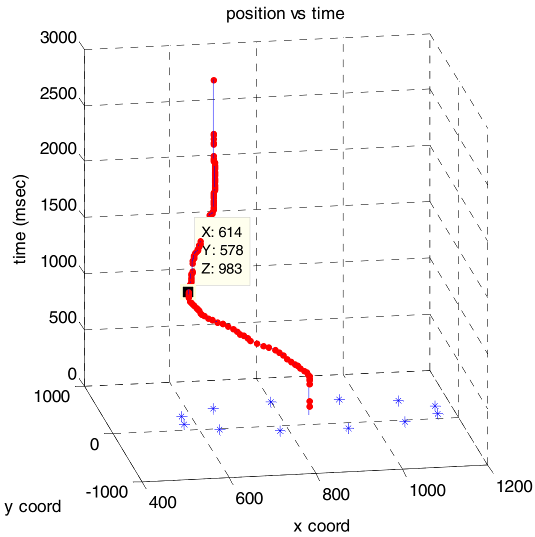

Fig. 4. Cursor trace shown in 3d with time on the z-axes. The location and timing of the correct decision is marked with a data tip.

5 Comparison

Target prediction with Murata's bearing algorithm and Kalman Filter perform similarly for able bodied users since the cursor trajectory follows a relatively simple path. Here Kalman Filter is more advantageous since keeping a history of data points is unnecessary. In addition, Kalman Filter also smoothes the cursor trajectories which is especially important for users with tremor. The noise characteristics can be personalized to specific users or can be adaptively updated.

For this paper, a total of 471 cases were processed. Both able-bodied and disabled users were present among subjects. Bearing algorithm failed to stay on correct target prediction at the end of the task in 35 cases whereas KA and KD failed in 2 and 1 cases respectively. KAD was able provide the correct prediction at the end of all tasks.

2 performance measures were considered for comparison: accuracy and sensitivity.

- Accuracy : ratio of correct predictions to all predictions
- Sensitivity: percent time or distance of correct prediction to indicate how quickly a decision was made

Prediction accuracy is obtained as 64% for Murata's algorithm and 65% for KAD. However it is possible to provide first correct prediction quicker with Kalman Filter and accuracy is increased for non-standard cursor traces. Kalman Filter framework does not provide better prediction for able-bodied users but requires less memory.

A decision firing mechanism can also be used for real-time systems in order to avoid target prediction update at every iteration as well as avoiding instant false detections. An easy choice is to look at consecutive decisions and check if the decision remains the same for several time instants. This could result in a no decision case in highly distorted cursor traces. Table 1 shows the availability of prediction in case of prediction at all iterations and firing mechanism where the same target was prediction at 3 consecutive time instants.

Table 1. Availability of prediction for bearing angle algorithm and Kalman Filter Framework

	bearing	KA	KD	KAD
prediction	471	470	470	471
Firing Mech.	446	455	459	469

6 Conclusion and Discussion

In this paper the possibility to use a simple process model and a Kalman Filter to predict a user's intended target was investigated. The results are promising to be used as target suggestions when a user is interacting with a system. However, the investigation was carried out in a restricted environment as an initial study. The performances should be investigated using more complex layouts and different input devices such as pointing devices. Process model could be developed as well as adaptive noise recognition. It is also possible to model the system such that the probabilities of targets will be directly obtained from Kalman Filter probability updates which will benefit more from process modeling.

The algorithms being developed will also benefit from analysis of the user interaction and acceptability where user has visual feedback of the intended target.

Acknowledgements. The authors would like to thank Dr.Pradipta Biswas of Engineering Design Centre for providing cursor data collection.

References

1. Murata, A.: Improvement of Pointing Time by Predicting Targets in Pointing With a PC Mouse. International Journal of Human-Computer Interaction 10, 23–32
2. Oirschot, H.K.-V., Houtsma, A.J.M.: Cursor trajectory analysis. In: Brewster, S., Murray-Smith, R. (eds.) Haptic HCI 2000. LNCS, vol. 2058, p. 127. Springer, Heidelberg (2001)

3. McGuffin, M.J., Balakrishnan, R.: Fitts' law and expanding targets: Experimental studies and designs for user interfaces. ACM Transactions Computer-Human Interaction 4, 388–422 (2005)
4. Lane, D.M., Peres, S.C., Sándor, A., Napier, H.A.: A Process for Anticipating and Executing Icon Selection in Graphical User Interfaces. International Journal of Human Computer Interaction 2, 243–254 (2005)
5. Wobbrock, J.O., Fogarty, J., Liu, S., Kimuro, S., Harada, S.: The Angle Mouse: Target-Agnostic Dynamic Gain Adjustment Based on Angular Deviation. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems (CHI 2009), New York, pp. 1401–1410 (2009)
6. Asano, T., Sharlin, E., Kitamura, Y., Takashima, K., Kishino, F.: Predictive Interaction Using the Delphian Desktop. In: Proceedings of the 186th Annual ACM Smposium on User Interface Software and Technology (UIST 2005), New York, pp. 133–141 (2005)
7. Lank, E., Cheng, Y.N., Ruiz, J.: Endpoint prediction using motion kinematics. In: Proceedings of the SIGCHI conference on Human factors in computing systems (CHI 2007), New York, NY, USA, pp. 637–646 (2007)
8. Keates, S., Hwang, F., Langdon, P., Clarkson, P.J., Robinson, P.: Cursor measures for motion-impaired computer users. In: Proceedings of the Fifth International ACM Conference on Assistive Technologies – ASSETS, New York, pp. 135–142 (2002)
9. Godsill, S., Vermaak, J.: Models And Algorithms For Tracking Using Variable Dimension Particle Filters. In: International Conference on Acoustics, Speech and Signal Processing (2004)
10. Hwang, F., Keates, S., Langdon, P., And Clarkson, P.J.: A submovement analysis of cursor trajectories. Behaviour and Information Technology 3(24), 205–217 (2005)
11. Langdon, P.M., Godsill, S., Clarkson, P.J.: Statistical Estimation of User's Interactions from Motion Impaired Cursor Use Data. In: 6th International Conference on Disability, Virtual Reality and Associated Technologies (ICDVRAT 2006), Esbjerg, Denmark (2006)
12. Zhang, X., Ren, X.: Optimizing Parameter Settings in Target Predictor for Pointing Tasks. In: International Conference Innovative Computing, Information and Control (ICICIC 2007) (2007)
13. Ziebart, B., Dey, A., Bagnell, J.A.: Probabilistic pointing target prediction via inverse optimal control. In: Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces (IUI 2012), pp. 1–10 (2012)