Trajectory prediction model for crossing-based target selection

Hao ZHANG¹, Jin HUANG¹*, Feng TIAN², Guozhong DAI¹, Hongan WANG¹,²

¹. Beijing Key Lab of Human-Computer Interaction, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China
². State Key Lab of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China

* Corresponding author, huangjin@iscas.ac.cn

Received: 26 February 2019   Accepted: 1 April 2019

Supported by National Key R&D Program of China (2016YFB1001405); the National Natural Science Foundation of China (61802379); Key Research Program of Frontier Sciences, CAS (QYZDY-SSW-JSC041).

Citation: Hao ZHANG, Jin HUANG, Feng TIAN, Guozhong DAI, Hongan WANG. Trajectory prediction model for crossing-based target selection. Virtual Reality & Intelligent Hardware, 2019, 1(3): 330—340
DOI: 10.3724/SP.J.2096-5796.2019.0017

Abstract   Background Crossing-based target selection motion may attain less error rates and higher interactive speed in some cases. Most of the research in target selection fields are focused on the analysis of the interaction results. Additionally, as trajectories play a much more important role in crossing-based target selection compared to the other interactive techniques, an ideal model for trajectories can help computer designers make predictions about interaction results during the process of target selection rather than at the end of the whole process.

Methods   In this paper, a trajectory prediction model for crossing-based target selection tasks is proposed by taking the reference of a dynamic model theory.

Results Simulation results demonstrate that our model performed well with regard to the prediction of trajectories, endpoints and hitting time for target-selection motion, and the average error of trajectories, endpoints and hitting time values were found to be 17.28%, 2.73mm and 11.50%, respectively.

Keywords   Target selection; Crossing-based selection; Trajectory prediction

1 Introduction

Target selection is a topic of significant interest in the human-computer interaction (HCI) community. It has been widely used in various continuous interaction spaces, such as the motion trajectories of virtual characters under users' control, and the selection of stationary and moving targets in AR or VR environment. Selection of targets by crossing a boundary "goal" instead of pointing inside a perimeter can, in some cases, achieve a higher efficiency compared with conventional interactive techniques. For example, crossing-based selection results in lower error rates and a higher interactive speed for a continuous crossing task with direction constraint. However, most research on target selection focuses on the analysis of interaction results—the endpoints of a trajectory. Only a few researchers in the HCI community have focused on the analysis of the trajectories themselves. A good understanding of the trajectories of target selection is important as it can provide insight and guidance on the effects of practice on performance, rational decision-making, and layout design in these user interfaces[1]. An ideal model for the entire target-selection motion process may help computer designers make predictions about interaction results during the process of target selection rather than at the end of the process.

It is a much more challenging job to model the trajectory for crossing-based selection than the
interaction results. Users tend to keep a comfortable angle, through which they can see the target they want to select during the process of approaching the targets\(^2\). Due to the elongated-rectangle target, users prefer to cross the target on its longer side, with the main impact on trajectory being the bending of the path\(^3\).

Control theory is commonly used to model trajectories for target selection tasks. However, the existing trajectory prediction model is not able to model the trajectory for crossing-based selection well due to the lack of consideration of the shape of the target. Moreover, control theory and dynamic models are usually used to simulate a reaching movement. Both models quantify the location and velocity at each instant according to the control signal being dynamically adjusted by the position and velocity feedback.

While referring to the same core ideas as existing trajectory models and correlation theories for dynamic models, our work contributes a new model for crossing-based interaction techniques. We find that our model achieved satisfactory simulation results on trajectory similarity, endpoints, and time fitting.

## 2 Related work

### 2.1 Crossing-based target selection

In the process of selecting targets, humans are able to strike a good balance between speed and accuracy. It has been shown that Fitts' Law\(^4\) is the most robust and successful model for human motion behavior, as it accurately predicts the time to complete target acquisition. An early crossing-based selection experiment referring to a "goal passing task" was performed by Accot and Zhai, which laid the foundation for the Steering Law\(^5\). Apitz et al.\(^6\) specified the six task conditions for indirect stylus input. Luo and Vogel\(^7\) found generalizable and empirical support for the application of crossing-based selection to touch input through analysis of the six task conditions.

Forlines et al.\(^8\) found that the average interaction time for crossing-based target selection is 16% faster than that for indirect stylus input. Taking into consideration the influence of the target's shape, Dixon et al.\(^9\) tested crossing target density and orientation. Using a direct stylus input device, they found that crossing-based target selection is faster than pointing for dialog boxes while also remaining spatially efficient. Cockburn et al.\(^10\) and Buxton et al.\(^11\) tested direct stylus input and indirect stylus input respectively, especially the friction force between finger and surface while dragging, and indicated that the input mode appeared to affect the performance of target selection. However, most basic research has focused on the duration and the endpoint of the task while no attention has been paid to the trajectory.

### 2.2 Trajectory prediction model

Apart from the research on endpoints and duration analysis of target selection, there has been a small amount of work on trajectory analysis. There have been attempts to explain the biological mechanism using the optimal control model\(^12\). In addition to predicting average behavior, the optimal control model can also simulate the feedback of unexpected changes in the real environment. This kind of model can therefore reflect the uncertainty, delay and unstable fluctuation during the process of selecting targets and adjust according to feedback.

Huang et al.\(^13\) built a target-selection motion model based on a linear-quadratic-Gaussian optimal feedback control (OFC) mechanism. The OFC model can simulate a static or moving circular target, while there is no thought for modeling the movement time. Quinn and Zhai\(^14\) developed a production model which can predict users' timing performance while typing using word-gesture keyboards. However, due to the specific application, other factors such as semantic information would be helpful to the model, and it
may be not suitable for general target selection tasks. In crossing-based target selection, the specifics of the target's shape results in a more curved path and more challenges to trajectory modeling, which is difficult for the existing model to simulate.

### 2.3 Dynamic models

A dynamic model can represent the behavior of an object over time, and is widely used in the field of dynamic simulations. A dynamic model has some similarities to the existing trajectory prediction model. Both of them calculate the moving direction and velocity of a particle in the current state, and the particle dynamics can be adjusted using feedback from both target and environment. Finally, the motion state of the particle can be calculated step by step.

Dynamic models have been used to model reaching movements of both humans and animals. Yekutieli et al.\cite{15} used a dynamic model to simulate the reaching movement of octopus arms. Tahara et al.\cite{16} constructed a musculoskeletal redundant arm model to simulate the reaching movement of human arms. However, target selection is a much more microscopic movement, where the environment and users' slight psychological fluctuations may lead to large changes of trajectory. Oulasvirta et al.\cite{17} attempted to use neuromechanics to model the process in which users press a button, which also provided a solution for more elaborate interactions. One case of dynamic models that had been applied in trajectory prediction is known as social force model (SFM). Helbing and Molnar first introduced the term of "social forces"\cite{19,21} and presented how they used them to simulate the motion of pedestrians by measuring the internal motivations of the individuals to perform certain movements\cite{19,21}.

In this paper, using dynamic model and SFM as a theoretical basis, the cursor under the users' control is regarded as a particle. Multiple factors which may influence the movement of the cursor are decomposed into mechanical concepts. Finally, the motion state and trajectory for crossing-based target selection can be predicted according to Newton's second law.

### 3 Trajectory prediction model

To model crossing-based target selection motion, we build a mechanical model referring to the relevant dynamic models. We approximate the movement of a pointing device by a particle being pushed with a controlled force, which can be described by Newton's second law\cite{18}. The main forces that affect the motion of particle $p$ are as follows:

#### 3.1 Desired force

In the process of approaching the target, a particle tends to reach a moving destination with minimum perturbations in velocity due to commands from the user. Referring to SFM\cite{19}, we introduce the concept of desired direction and desired velocity. The particle $p$ of mass $m_p$ tends to move with a desired speed $v_p^0$ towards the moving target with a changeable direction vector $\overrightarrow{e_{ps}}(t)$, and therefore the particle is likely to correspondingly alter its actual velocity $v(t)$ with a relaxation time $\tau_p$. The desired force of the particle can be described by an acceleration term of the form:

$$f_d(t) = \frac{v_p^0 \overrightarrow{e_{ps}}(t) - v(t)}{\tau_p}$$

\hspace{1cm} (1)

#### 3.2 Inertial losses

During the process of approaching the target, inertia exists which may influence the subsequent frames of
motion trajectory in the case of an accelerated motion. At the beginning of the motion, the user tries to hit the target in a specific time, and the motion process can be considered as an impulse motion. The inertial action will decrease as the particle gradually approaches the target, because a relatively lower velocity is needed in order for users to select the moving target\(^{(20)}\). For this reason, we introduce a linear inertial loss function to simulate the phenomenon of inertial losses. It is given by:

\[
    f_{\text{inertial}}(t) = w^k \frac{v^\theta_p e_{pm}(t) - v(t)}{\tau_p}
\]

(2)

The coefficient of inertial losses \(w^k\) is:

\[
    w^k = w_{\text{ini}} - \frac{w_{\text{ini}} - w_{\text{end}}}{2k_{\text{max}}} k_i
\]

(3)

where \(w_{\text{ini}}\) and \(w_{\text{end}}\) are system parameters, \(k_{\text{max}}\) is the maximum number of trajectory's frames, \(k_i\) is the current frame of the motion process.

### 3.3 Boundary force

The particle also tends to keep a certain distance from the border of the screen. Users may feel uncomfortable as the stylus or fingers moves towards the screen boundary. Therefore, the boundary of the screen has a repulsive effect that can be described by:

\[
    f_{\text{br}}(t) = \left[ A \exp \left( \frac{r_p}{B} \right) \right] \overline{n}_{\text{br}}
\]

(4)

where \(\overline{n}_{\text{br}}\) is the unit vector denotes the direction perpendicular to boundary, \(r_p\) shows the distance between particle and boundary, and \(A, B\) are system parameters.

### 3.4 Interactive force

The targets used in crossing-based target selection are generally elongated rectangles, and users customarily aim at the longer side to cross the target. In this case, we modify the general moving direction and the insertion angle when crossing the target. As suggested in the literature\(^{(21)}\), we use angle to control both the bending of the trajectory and the insertion angle as influenced by psychological factors.

The intersection angle between moving direction of particle \(e_{j}(t)\) and target \(e_{a}(t)\) is used to control the bending of the path. The intersection angle \(\theta_i\) and the force \(f_{i}(t)\) used to control direction of movement are defined as:

\[
    \theta_i = \langle e_{j}(t), e_{a}(t) \rangle
\]

(5)

\[
    f_{i}(t) = G \exp \left[ \frac{D(\theta_i)^2 r_p}{F} \right]
\]

(6)

A reference intersection angle \(\theta_i\) and the force \(f_{i}(t)\) to control inserted angle are given by:

\[
    \theta_i = \langle e_{ja}(t), n_{\text{nor}} \rangle
\]

(7)

\[
    f_{i}(t) = G \exp \left[ \frac{H(\theta_i)^2 r_p}{F} \right]
\]

(8)

where \(n_{\text{nor}}\) is the normal vector of orientation of target, \(e_{ja}(t)\) is the unit vector denoting the direction from the particle to the target, \(r_p\) is the distance between target and particle.

Referring to SFM and the Newton’s second law, the model for crossing-based target selection motion is finally defined by:
\[ m_p \frac{dv_p}{dt} = m_p f_o(t) + f_{\text{stance}}(t) + f_{\text{lift}}(t) + f_{\text{g}}(t) \] (9)

4 Experiment design

To generate empirical data for estimating the parameters, we conducted a crossing-based selection task referring to the work of Luo and Vogel\(^7\).

4.1 Tasks

We designed similar tasks to those used by Luo and Vogel\(^7\). Participants used a mouse to cross a moving target with one of two different orientations (Ori), one of eight directions of movement (Dir), width of 96 pixels and a velocity of 192 pixels per second. The initial distance between target and start point was 960 pixels. The crossing-based target selection tasks are illustrated in Figure 1.

4.2 Participants and apparatus

We recruited 15 participants (8 females and 7 males, with an average age of 27.3) in this study. All of them were familiar with computer and stylus use. The experiment was conducted on a Lenovo ThinkPad X1 laptop computer, with an Intel Core i7 8550 CPU at 1.8GHz, connected to a Wacom pen display with a stylus. The pen display was a direct interactive screen that can only be operated by stylus, 13.3 inches in size and with 1920×1080 pixels resolution. The stylus was 15.4cm in length, 9mm in diameter at the barrel, and 10g in weight. The system ran with a sampling frequency of 100Hz. The experiment programs in this study were developed using Unity3D with C# code. The experimental equipment and the experimental apparatus can be seen in Figure 2.

4.3 Optimization parameters

The system parameter set \([A, B, C, D, E, F, G, H, I, w_m, w_{\text{ext}}, \theta_r, \theta]\) is selected according to the model defined in Section 3, which may significantly affect the similarity between the real and simulated data. We defined the sum total Euler distance per frame as the cost function in order to estimate the similarity value. We developed a genetic algorithm to obtain the most appropriate parameter set. With the population size set at 50, the optimization parameter set is shown in Table 1.
5 Simulation results

To simulate the trajectory for a task situation, we initialized the cursor at a mean state of $p_0 = \left[ p_x(t_0); p_y(t_0); \dot{p}_x(t_0); \dot{p}_y(t_0); \tau_0; v_p^0 \right]$ where $(p_x(t_0), p_y(t_0))$ is the start position of the cursor, $(\dot{p}_x(t_0), \dot{p}_y(t_0))$ is the initial moving direction, $\tau_0$ is the relaxation time which shows the time required for the cursor to move from a state of larger velocity fluctuations to a stable state during the process of selecting the target, and $v_p^0$ is the scalar quantity of desired velocity. The moving target $\alpha$'s position was set at $(1120, 540)$. The empirical parameters are shown in Table 2.

### 5.1 Comparison of trajectories

Using the proposed trajectory prediction model and the optimization parameters calculated from the empirical data, we modeled the trajectories for crossing-based target selection. The motion process began at the time of crossing the start rectangle and ended when the moving target was hit or the cursor exited the screen. The comparison of trajectories is shown in Figure 3.

In order to quantitatively analyze the similarity of the trajectories, we use the Euclidean distance in each frame between the real data and the simulated data. The similarity statistics of 16 composite cases of 8 Dir and 2 Ori are as shown in Figure 4.

As we can see from Figure 3, when compared with the real trajectories, the predicted trajectories have a

### Table 1 Optimization parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1847.5</td>
</tr>
<tr>
<td>B</td>
<td>1157.4</td>
</tr>
<tr>
<td>C</td>
<td>$-2099.5, &lt;e_p(t), e_n(t)&gt; &lt; \theta_c$</td>
</tr>
<tr>
<td>D</td>
<td>$-0.4793, &lt;e_p(t), e_n(t)&gt; &lt; \theta_c$</td>
</tr>
<tr>
<td>E</td>
<td>$4.9962, &lt;e_p(t), e_n(t)&gt; &lt; \theta_c$</td>
</tr>
<tr>
<td>F</td>
<td>1220.3</td>
</tr>
<tr>
<td>G</td>
<td>$-1282.4, &lt;e_p(t), n_{\text{ori}}^\alpha &gt; &lt; \theta_l$</td>
</tr>
<tr>
<td>H</td>
<td>$2.1322, &lt;e_p(t), n_{\text{ori}}^\alpha &gt; &lt; \theta_l$</td>
</tr>
<tr>
<td>I</td>
<td>$0.4991, &lt;e_p(t), n_{\text{ori}}^\alpha &gt; &lt; \theta_l$</td>
</tr>
</tbody>
</table>

| $w_{\text{ori}}$ | 0.9 |
| $w_{\text{val}}$ | 0.7 |
| $\theta_c$      | 0.1948 |
| $\theta_l$      | -0.1907 |

### Table 2 Empirical parameters

<table>
<thead>
<tr>
<th>Ori $(^\circ)$</th>
<th>Dir $(^\circ)$</th>
<th>$(p_x(t_0), p_y(t_0))$</th>
<th>$(\dot{p}_x(t_0), \dot{p}_y(t_0))$</th>
<th>$\tau_0$</th>
<th>$v_p^0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-135</td>
<td>(841.8, -380.5)</td>
<td>(162.7, 522.1)</td>
<td>0.045</td>
<td>1558.4</td>
</tr>
<tr>
<td>0</td>
<td>-90</td>
<td>(881.7, -335.2)</td>
<td>(167.1, 526.7)</td>
<td>0.029</td>
<td>1542.6</td>
</tr>
<tr>
<td>0</td>
<td>-45</td>
<td>(873.2, -634.5)</td>
<td>(164.3, 516.9)</td>
<td>0.039</td>
<td>1785.6</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>(853.2, -525.0)</td>
<td>(162.6, 522.7)</td>
<td>0.020</td>
<td>1734.6</td>
</tr>
<tr>
<td>0</td>
<td>45</td>
<td>(897.9, -403.3)</td>
<td>(166.4, 530.2)</td>
<td>0.069</td>
<td>1734.3</td>
</tr>
<tr>
<td>0</td>
<td>90</td>
<td>(869.0, -627.1)</td>
<td>(165.3, 518.8)</td>
<td>0.034</td>
<td>1712.3</td>
</tr>
<tr>
<td>0</td>
<td>135</td>
<td>(929.0, -434.1)</td>
<td>(166.5, 520.0)</td>
<td>0.015</td>
<td>1580.3</td>
</tr>
<tr>
<td>0</td>
<td>180</td>
<td>(830.1, -305.7)</td>
<td>(164.2, 526.1)</td>
<td>0.023</td>
<td>1508.0</td>
</tr>
<tr>
<td>90</td>
<td>-135</td>
<td>(1333.0, 42.8)</td>
<td>(188.7, 543.4)</td>
<td>0.027</td>
<td>1540.6</td>
</tr>
<tr>
<td>90</td>
<td>-90</td>
<td>(1311.6, 48.3)</td>
<td>(189.5, 545.1)</td>
<td>0.025</td>
<td>1535.1</td>
</tr>
<tr>
<td>90</td>
<td>-45</td>
<td>(1350.2, 59.5)</td>
<td>(184.0, 541.5)</td>
<td>0.040</td>
<td>1765.5</td>
</tr>
<tr>
<td>90</td>
<td>0</td>
<td>(1369.9, 54.0)</td>
<td>(190.0, 545.8)</td>
<td>0.020</td>
<td>1808.6</td>
</tr>
<tr>
<td>90</td>
<td>45</td>
<td>(1292.5, 38.6)</td>
<td>(187.0, 544.7)</td>
<td>0.026</td>
<td>1704.9</td>
</tr>
<tr>
<td>90</td>
<td>90</td>
<td>(1401.8, 19.0)</td>
<td>(191.6, 542.0)</td>
<td>0.027</td>
<td>1669.5</td>
</tr>
<tr>
<td>90</td>
<td>135</td>
<td>(1322.2, 82.8)</td>
<td>(187.5, 546.5)</td>
<td>0.028</td>
<td>1534.6</td>
</tr>
<tr>
<td>90</td>
<td>180</td>
<td>(1285.3, 7.8)</td>
<td>(186.9, 545.2)</td>
<td>0.051</td>
<td>1521.4</td>
</tr>
</tbody>
</table>
shorter overall length and greater similarity when \( \text{Ori} = 90^\circ \). As shown in Figure 4, the overall trajectory is relatively longer when \( \text{Ori} = 0^\circ \) than when \( \text{Ori} = 90^\circ \), and the average length of trajectory is 7362.33 pixel, with a relatively small deviation of 15.02%. The average length of trajectory is 5092.20 pixel when \( \text{Ori} = 90^\circ \), and the deviation of the trajectory is higher at 19.55%, while the overall deviation ratio is 17.28%.

![Comparison of trajectories.](image1)

![Comparison of trajectory’s similarity.](image2)

**Figure 3** Comparison of trajectories.

**Figure 4** Comparison of trajectory’s similarity.

### 5.2 Results of endpoints

The trajectory simulation results in Section 5.1 show that the proposed model provides good results on the prediction of endpoints. A comparison of endpoints can be seen in Table 3. Results show that the average error of the mean endpoint is 2.73 mm, while the total length of the target is 14.4 mm. Different orientations have a small impact on the errors in the endpoint. When \( \text{Ori} = 0^\circ \), the average error in the mean endpoint is...
3.09mm, while the average errors of the mean endpoint is 2.36mm when Ori = 90°. The maximum error in the endpoint is 4.31mm when Ori = 90°, Dir = 180°, while the minimum error is 1.23mm when Ori=90°, Dir=45°.

5.3 Results of time fitting

We further studied a comparison of time fitting between the empirical data and the predicted data. The ratio of the predicted time to the empirical data is used as the criterion to measure the similarity. The time fitting results are shown in Figure 5.

As shown, the model proposed in our paper has a good prediction for the time fitting of crossing-based target selection, with an average error of 11.5%. Simulation results show that the time is better-estimated when Ori = 90° with an average error of 6.92%, and variability in the error of 16.07%. The minimum value of the error occurs when Ori = 90°, Dir = 0° (2.95%), while the maximum value is 24.17% when Ori = 0°, Dir = 135°.

![Figure 5 Comparison of time fitting.](image)

<table>
<thead>
<tr>
<th>Ori (°)</th>
<th>Dir (°)</th>
<th>Offset of Endpoints (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-135</td>
<td>4.13</td>
</tr>
<tr>
<td>0</td>
<td>-90</td>
<td>3.52</td>
</tr>
<tr>
<td>0</td>
<td>-45</td>
<td>2.18</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>3.30</td>
</tr>
<tr>
<td>0</td>
<td>45</td>
<td>1.94</td>
</tr>
<tr>
<td>0</td>
<td>90</td>
<td>3.03</td>
</tr>
<tr>
<td>0</td>
<td>135</td>
<td>2.95</td>
</tr>
<tr>
<td>90</td>
<td>-135</td>
<td>2.30</td>
</tr>
<tr>
<td>90</td>
<td>-90</td>
<td>1.41</td>
</tr>
<tr>
<td>90</td>
<td>-45</td>
<td>1.98</td>
</tr>
<tr>
<td>90</td>
<td>0</td>
<td>3.10</td>
</tr>
<tr>
<td>90</td>
<td>45</td>
<td>1.23</td>
</tr>
<tr>
<td>90</td>
<td>90</td>
<td>1.75</td>
</tr>
<tr>
<td>90</td>
<td>135</td>
<td>2.84</td>
</tr>
<tr>
<td>90</td>
<td>180</td>
<td>4.31</td>
</tr>
</tbody>
</table>

6 Discussion

6.1 Curved paths

From the actual trajectories obtained from user experiments, it is found that the process of target selection can be divided into two parts: the impulse motion process to approach the target, and the corrective motion process to land on the target. In the first part of the motion process, the user adjusts the direction of motion so that they can use the target as visual feedback. In the second part of the motion process, the user will adjust the angle of crossing the target appropriately to ensure the correct rate of target selection. The desire
to gain visual feedback and maintain a comfortable angle in user's mind led to the curved paths.

6.2 Influence of orientation

When Ori = 90°, the moving direction of the cursor is perpendicular to the long side of the target, and users do not need to adjust the direction to ensure a proper angle of crossing. However, users need to adjust the direction of movement twice when Ori = 0°, first for visual feedback while approaching target, and second to achieve the proper angle to cross the target. As we can see from the empirical data, the intersection angle of crossing the target is almost perpendicular to the target if there is enough reaction time (e.g., when the moving direction of the target is 90°). The adjustments mentioned all require additional control, which may affect the overall speed of selecting the target and lead to a more curved path.

6.3 Dynamic model for target selection

In this paper, we use a dynamic model to simulate the process of selecting a moving target under users' control. Mechanical principles are used to model how users think and move while selecting targets. We regarded the cursor as a particle, and quantified the factors which may affect the movement of the particle using mechanical formulas: desire force is used to simulate users' psychological desire to approaching the target; inertia loss is used to model the transformation from impulse motion to corrective motion; boundary force is used to simulate the influence of the scene boundary while moving towards the target; intersection angles are used to control the direction of motion to model the curve of the path. The motion state of the particle can be calculated and updated by reference to Newton's second law. At the same time, the success of a dynamic model in relating to emotions, multiple users and other factors may provide guidance to better understand the process of target selection.

6.4 Model evaluation

No attention was paid to the influence of the target's shape, so the existing trajectory prediction models could not accurately model the process of crossing-based target selection. In this paper, inspired by the relevant theory of dynamics, corrective forces and angle are used to simulate the curve of the path caused by the target's shape. Simulation results show that our model performed well in the prediction of the trajectories, endpoints and time to hit the target for target-selection motion, with the average errors of trajectories, endpoints and hitting time values being 17.28%, 2.73mm and 11.50% respectively.

The trajectory of crossing-based target selection has a shorter path length, shorter corrective adjustments, and shorter interactive time when Ori = 90°, compared with the trajectory when Ori = 0°. In addition, the model proposed in the paper performs better when Ori = 90° than when Ori = 0°.

6.5 Design in interfaces

In specific applications for crossing-based target selection, interface designers should avoid large curves of trajectory. To achieve this, designers should make the long side of the target perpendicular to the line connecting the starting point and the center of the target in order to increase the accuracy of target selection.

7 Conclusion and future work

In some interactive scenarios, crossing-based target selection motion may result in lower error rates and a
higher interactive speed. At present, most research in the field of target selection is focused on analysis of the time taken to reach the target and on interactive results. A good understanding of the trajectories of target selection is important as it can provide insight and guidance on the effects of practice on performance, rational decision-making, and layout design in these user interfaces. The main challenge for trajectory modeling is how to model the curve of the trajectory caused by user's reactions in a more appropriate way.

In consideration of the above problems and referring to the classical crossing-based target selection experimental setting, we recruited participants to carry out experiments. As we can see from the empirical trajectories, users prefer a good balance between speed and accuracy in the process of selecting targets, which User reactions will lead to more curved paths. Experiments show that users prefer to select the long side of the target to cross. Based on dynamic models, we proposed a trajectory prediction model for crossing-based target selection. Simulation results show that our model performed well in the prediction of trajectories, endpoints and hitting time for crossing-based target selection.

However, our model is still far from perfect. In the future, we will further analyze the impact of other parameters on crossing-based target selection. At the same time, some other influencing forces could be added into our model, such as motion uncertainty. Furthermore, it is possible to simplify the empirical data set. Our work may provide new perspectives for understanding target selection motion and other HCI research.

References

2 Accot J. Les tâches trajectorielles en interaction homme-machine: cas des tâches de navigation. Toulouse 1, 2001
10 Cockburn A, Ahlström D, Gutwin C. Understanding performance in touch selections: Tap, drag and radial pointing drag


