
Hands Free Mouse: Comparative Study on Mouse Clicks Controlled by Humming

Ondřej Poláček

Faculty of Electrical Engineering,
Czech Technical University in
Prague
Karlovo nám. 13
12135 Praha 2
Czech Republic
polacond@fel.cvut.cz

Zdeněk Míkovec

Faculty of Electrical Engineering,
Czech Technical University in
Prague
Karlovo nám. 13
12135 Praha 2
Czech Republic
xmikovec@fel.cvut.cz

Abstract

In this paper we present a novel method of simulating mouse clicks while the cursor is navigated by head movements tracked by webcam. Our method is based on simple hummed voice commands. It is fast, language independent and provides full control of common mouse buttons. Our method was compared with other three different methods in an experiment that proved its efficiency by means of task duration.

Keywords

Non-Verbal Vocal Interface, Voice Interface, Head Tracking, Accessibility, Comparative Study

ACM Classification Keywords

H.5.2 Information interfaces and presentation: User Interfaces – Input devices and strategies; Voice I/O.

General Terms

Design, Experimentation, Measurement, Performance

Introduction

Complex graphical user interfaces (GUI) are present not only in desktop computers, but they also appear in other areas such as Rich Internet Applications (RIA) on the Internet. The efficiency of interaction with such complex GUI is strongly dependent on the efficiency of

the way the mouse is used. For users with limited motor abilities (especially upper-limb impaired users) the use of mouse could be a serious problem. These users need an alternative way of mouse control. There are for instance several head tracking solutions which can solve the problem of cursor control satisfactorily, but not the problem of simulation of mouse clicks (see State of the Art). The set of the mouse clicks simulated is either not complete or too complicated for the user who cannot simulate mouse in full extent.

State of the Art

Several methods have been used to simulate mouse clicks. The camera mouse system [1] used a *dwelling time* method. A mouse click was generated, when the user kept the mouse cursor within a 30-pixel radius for 0.5 s. The dwelling time method is capable of simulating left click only and does not cope with other mouse events such as right click, double click, dragging and scrolling. Moreover the method raises the *Midas touch problem* [4], as the user cannot stop the cursor without issuing a left click. It can be solved either by adding places where the user can stop the cursor [1] or displaying a pop-up menu after the dwelling time expires [11].

Tracking various face features can be also used to simulate mouse clicks. For instance, a system published by Tu [10] responded to the state of user's mouth. It was capable of simulating left click by opening the mouth and right click by stretching the mouth. Dragging was provided by moving the cursor while keeping the mouth open. Another system called hMouse [2] triggered left and right clicks when the turn of user's head exceeded a specific threshold angle.

There was no solution for dragging and scrolling reported.

There are also several multimodal systems that use different interaction channel to simulate mouse clicks. Nouse [3] employed a computer keyboard, which is a rapid and complex solution, however, the users still need to use their hands and such solution cannot be used for disabled people with severe upper-limb impairment. Another modality that can be used for simulating mouse clicks is speech. Multimodal system published by Loewenich and Maire [5] defined five simple speech commands (*click, double, right, hold and drop*) that covered all clicking and dragging operations. Ronzhin and Karpov [8] published similar system that defined 30 speech commands covering clicking, dragging and scrolling one by one line. Remaining commands were used as shortcuts to common operations such as *open a file, exit an application, copy, paste* etc.

Our Solution

In our system *non-verbal vocal interaction* (NVVI) [6, 9] is used for simulating mouse clicks. This interaction method can be characterized as using other sounds than speech, such as humming, to control user interfaces. In our case, hummed voice commands are determined by its pitch and length. Expected pitch profiles of the commands are depicted in Figure 1. Left click (1a) is defined as a short tone produced below user-specific threshold pitch. Double click (1b) is defined as two consecutive left clicks. Right click (1c) is a short tone above the threshold pitch. Drag (1d) is a long tone. The difference between long and short tone is 0.5 s. However, this value can be modified according to preferences of the user. Drop operation does not

have its own command and it is triggered by short or long tone. Scrolling (1e, 1f) is performed when significant increase or decrease in pitch is detected. Amount of lines scrolled is determined by length of the voice command in real time. Continuous real-time control is a significant advantage of NVVI [9]. Using speech the user has to explicitly specify the amount of lines scrolled, which is rather awkward. Note that in order to keep our method simple, minimal amount of voice commands was used. Moreover, commands are very simple and short for the most frequent operations (clicking) and they are a bit more complicated for advanced operations (dragging and scrolling).

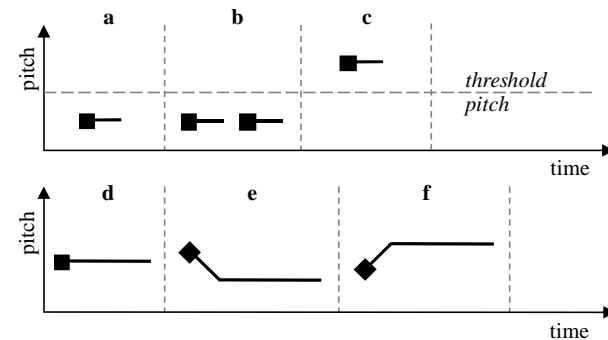


Figure 1. Non-verbal vocal commands used to simulate mouse clicks. **a.** left click, **b.** double click, **c.** right click, **d.** drag, **e.** scroll down, **f.** scroll up

This method is very well suited for real-time control, as the hummed commands are recognized much faster than verbal commands [9]. They are also culturally and language independent. On the other hand this is a very unusual way of interaction and the users have to get used to it [6].

Experiment

The aim of the experiment was to determine the efficiency of our solution. We compared our NVVI method with other three different methods for simulating mouse clicks in terms of speed and error rate. The following four methods were prepared for the comparison test:

- *Non-verbal vocal interaction (NVVI)* as described in previous section. Scrolling voice commands were not included in the experiment.
- *Speech commands.* Regarding the fact that all participants were Czech native speakers we used Czech commands recognized by MyVoice application [7]. Commands and their English equivalents are listed in Table 1.

Table 1. Speech commands.

Command in Czech	English Equivalent	Description (mouse operation)
Klik	Click	Left click
Dvojklik	Double Click	Double left click
Pravý klik	Right click	Right click
Vzít	Drag	Left button down
Položit	Drop	Left button up

- *Computer keyboard.* Mouse buttons were mapped to keystrokes. Alt + left arrow corresponded to left mouse button and alt + right arrow to right button. Arrows up and down corresponded to mouse wheel. This method was chosen as a reference test.
- *Head gestures.* This solution combined the dwell time approach with a pie menu (see Figure 2). When the mouse cursor did not significantly move for 0.5 s, the pie menu appeared and concrete operation was chosen by moving the cursor over the menu as

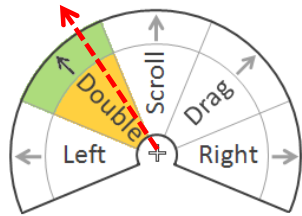


Figure 2. Pie menu.

depicted in Figure 2 by dashed arrow. This method does not suffer from Midas touch problem [4] and all mouse operations can be simulated. The menu can be cancelled by moving the cursor down.

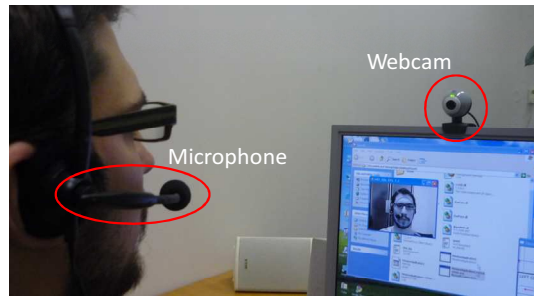


Figure 3. Setup of the experiment.

In order to navigate the cursor by head throughout the experiment, a head tracking system described in [11] was chosen. A cheap webcam can be used by the system to track the head of the user and convert its position and turn to position of the mouse cursor on a screen. An absolute mode is used as described in [5], i.e. the position of head is directly mapped to the position of the mouse cursor. A 17" LCD monitor with native resolution 1280 x 1024 pixels was used. There was a webcam mounted on the top of the monitor that provided data for head tracking. While using vocal modalities the participants used headphones with microphone. The experiment setup is depicted in Figure 3.

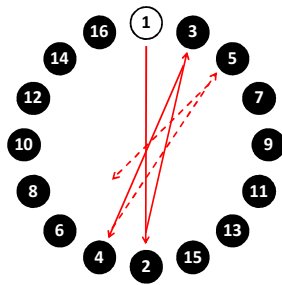


Figure 4. Task template.

For the experiment four mouse click simulation tasks were defined. In every task the participants had to move the cursor to particular circle as shown in Figure 4 and perform specified clicking operation. The

participants had to start with circle 1 and continue until circle 16 was reached. Part of expected cursor trajectory is shown by arrows. Every task defined a different mouse operation to be performed in the circles as follows:

- Task *Pointing*. No clicking was involved. This task was included for reference purposes.
- Task *Left Click*. Only left click had to be simulated.
- Task *Multi Click*. Left, right and double clicks were simulated according to caption of circles.
- Task *Drag & Drop*. Drag and drop operations were involved.

The participants had to pass through overall 16 tasks (four tasks using four modalities for simulating mouse clicks). In order to minimize learning effect, the sequence of methods and tasks was shuffled. Moreover every task had to be undertaken twice and data were measured only in the second try. The objective data collected were processed into three indicators as follows:

- *Task duration*, which is the duration between the first and last operation in a task including error operations. This indicator is used to measure the efficiency of each method.
- *Click duration*, which is the duration between passing the border of a small circle and a correct click.
- *Error rate*, which expresses the number of wrong clicks relative to number of total clicks.

Due to the long-lasting single session (about 50 minutes), we did not include scrolling capabilities of

Table 2. Mean times and standard deviations (SD) for each task and modality. Grey cells in one row correspond to means that are not statistically different. Overall error rates for each method are shown in the last row.

		Speech		NVVI		Keyboard		Head gestures	
		Mean Time [s]	SD [s]	Mean Time [s]	SD [s]	Mean Time [s]	SD [s]	Mean Time [s]	SD [s]
Left	Task Duration	49.6	7.4	39.0	7.8	34.0	6.5	60.9	9.7
Click	Click Duration	1.342	0.277	0.875	0.203	0.524	0.175	1.927	0.260
Multi	Task Duration	55.1	7.7	49.5	12.9	39.3	7.0	71.1	14.3
Click	Click Duration	1.639	0.374	1.358	0.526	0.706	0.213	1.977	0.415
Drag &	Task Duration	51.0	6.7	44.5	8.2	36.3	6.8	70.5	9.9
Drop	Click Duration	1.430	0.274	1.256	0.249	0.558	0.186	1.667	0.314
Error rate [%]		3.53		6.11		4.09		1.75	

evaluated methods. After each session the participants were given a post-test questionnaire to subjectively assess speed, comfort and accuracy of each method. In the experiment 54 participants without disabilities took part. They were recruited from university students (mean age=23.5, SD=0.98) and were technically oriented and experienced computer users. The participants were trained to perform NVVI and head gestures in a training session which was conducted before the experiment and lasted approximately 15 minutes. Speech and keyboard methods were not trained because speech is a natural form of interaction and participants were experienced enough in using keyboard.

Results

In Table 2 the results of the experiment are summarized. Mean times (task and click durations) of each method in three tasks are shown in rows. As these times are compared in each row, the speed of the methods can be evaluated without exception as follows:

Keyboard < NVVI < Speech < Head gestures

ANOVA test and Scheffé's method were used to find statistically significant ($p < .01$) differences in mean times of each task. Most of them are significant except those shown in grey color in Table 2. Our method is the fastest among hands-free methods (speech and head gestures), however, it is slower than keyboard, which on the other hand is unusable for severe motor impaired users.

Error rate results are summarized in the last row of Table 2. The head gestures method experienced the lowest error rate. This is probably caused by the relatively high time penalty, when the user selects a wrong option. In this case the cursor has to be navigated to the initial position and the user has to wait for pie menu popup (dwell time). This leads to much more careful interaction. However, we believe, that this behavior can be improved by personalizing the dwell time and size of the pie menu. The error rate of NVVI was the highest one (6.11%), which is caused by insufficient training involving only one session. According to longitudinal studies [6, 9] four training sessions are enough to minimize error rate of the NVVI.

Nevertheless the time penalty caused by these errors is already included in the task duration indicator.

Table 3. Questionnaire results. Scale 1 (=worst) ... 5 (=best). Mean values are displayed.

	Speech	NVVI	Kbd.	Gest.
Speed	3.33	3.50	4.72	2.11
Comfort	3.74	2.81	4.30	2.89
Accuracy	3.94	2.94	4.81	3.04

Subjective results are shown in Table 3. Head gestures were generally rated by the users as the worst method and keyboard as the best. Even though NVVI was faster than speech, it was perceived worse in comfort and accuracy.

Conclusion

In this paper we have described a method for mouse clicks simulation based on humming (NVVI). This method is capable of simulating all common mouse buttons including mouse wheel for real-time scrolling. Our method was compared with other three methods (speech, head gestures and keyboard) and it was the second fastest, although it experienced the highest error rate. The subjective perception of the accuracy and comfort was also rated as the worst. In the future, we will conduct longitudinal tests with disabled users in real applications and combine more modalities in the system to provide more efficient control of a computer.

Acknowledgement

We would like to thank Lukáš Zich from the Center for Machine Perception, CTU in Prague for provision of the head tracking software. This research has been partially supported by the MSMT research program MSM 6840770014 and the VitalMind project (IST-215387).

References

- [1] Betke, M., Gips, J. and Fleming, P. The Camera Mouse: Visual Tracking of Body Features to Provide Computer Access for People With Severe Disabilities. In *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, IEEE Computer Society (2002), 1-10.
- [2] Fu, Y., Huang, T. S., hMouse: Head Tracking Driven Virtual Computer Mouse, In *Proc WACV'07*, IEEE Computer Society (2007), 30-36.
- [3] Gorodnichy, D. O., Malik, S. and Roth, G. Nouse 'Use Your Nose as a Mouse' - a New Technology for hands-free Gamers and Interfaces, In *Proc VI'2002*, Calgary (2002), 354-361.
- [4] Jacob, R. J.K. What you look at is what you get, In *Computer*, vol.26, no.7, IEEE Computer Society (1993), 65-66.
- [5] Loewenich, F. and Maire, F. Hands-free mouse-pointer manipulation using motion-tracking and speech recognition. In *Proc OZCHI*, ACM Press (2007), 295-302.
- [6] Mahmud, M., Sporka, A. J., Kurniawan, S. H. and Slavik, P. A Comparative Longitudinal Study of Non-verbal Mouse Pointer, In *Proc INTERACT 2007*, Springer-Verlag (2007), 489-502.
- [7] Nouza, J., Nouza, T. and Červa, P. A Multi-Functional Voice-Control Aid for Disabled Persons. In *Proc SPECOM 2005*, Moscow, 715-718.
- [8] Ronzhin, A. and Karpov, A. Assistive multimodal system based on speech recognition and head tracking, In *Proc of EUSIPCO'2005*, Turkey, 2005.
- [9] Sporka, A. J., Kurniawan, S. H., Mahmud, M. and Slavik, P. Longitudinal study of continuous non-speech operated mouse pointer, In *Proc CHI'07*, ACM Press (2007), 2669-2674.
- [10] Tu, J., Huang, T. and Tao, H. Face as Mouse Through Visual Face Tracking, In *Proc CRV* (2005), IEEE Computer Society (2005), 339-346.
- [11] Zich, L. Video based Human-Computer interface, Master Thesis, 2009, CTU Prague, FEE.