# Multi-flick: An Evaluation of Flick-Based Scrolling Techniques for Pen Interfaces

**Dzimitry Aliakseyeu**
Media Interaction
Philips Research
The Netherlands
dzmitry.aliakseyeu@philips.com

**Pourang Irani**
Computer Science
Univ of Manitoba
Canada
irani@cs.manitoba.ca

**Andrés Lucero**
Industrial Design
Eindhoven Univ of Tech
The Netherlands
a.a.lucero@tue.nl

**Sriram Subramanian**
Computer Science
Univ of Bristol
U.K.
sriram@cs.bris.ac.uk

**ABSTRACT**

*Multi-flick*, which consists of repeated flick actions, has received media attention as an intuitive and natural document-scrolling technique for stylus based systems. In this paper we put multi-flick to test, by designing several flick-based scrolling techniques. We map out the design space of multi-flick and identify mapping functions that make multi-flick a natural and intuitive technique for document navigation. We then compare several multi-flick variations for navigating lists on three different devices – a PDA, a tabletPC, and a large table. Our study shows that compound-multi-flick (CMF) is the most preferred technique and it is at least as fast, if not faster than the traditional scrollbar. In a follow-up study, we evaluate multi-flick for scrolling text-based documents. Results show that all implementations of multi-flick are as good as the scrollbar for short distances while CMF is the most preferred. We discuss the implications of our findings and present several design guidelines.

**ACM Classification:** H5.2 [Information interfaces and presentation]: User Interfaces - Graphical user interfaces.

**General terms:** Design**,** Human Factors

**Author Keywords**

Scrolling, Flick, Multi-flick, Tables, Tablets.

**INTRODUCTION**

Researchers are devising methods for facilitating basic tasks such as pointing, selecting, and scrolling to make interactive pen-based environments more accessible to users. Scrolling is an important interaction technique for supporting many document related daily tasks. Scrolling shifts into the viewport parts of a document that are of interest and that reside off-screen. Bryne et al. [5] found that users spend 13.4% of a five-hour web-browsing session simply scrolling documents. Therefore any small improvements to scrolling can result in significant benefits to users.

While displacement and rate-based systems have been effective on mouse-wheels or isometric joysticks, such techniques are less effective for scrolling documents with pen-

based devices. For instance, the scrollbar is impractical on a tabletop or on a large display because, as display size increases, the physical effort and reach required for scrolling interactions increases correspondingly. As a result, there is a need for producing more natural forms of interactive navigation tools for pen-based systems.
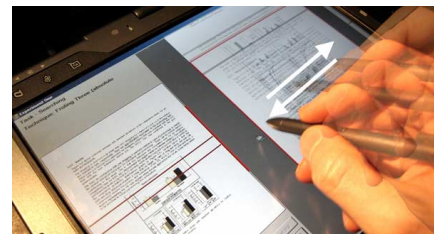


**Figure 1: (best in color) Multi-flick scrolling technique.**

Recently, designers and researchers have introduced *flicking* as an intuitive and natural form for throwing objects in virtual environments [23] and for shifting content within a viewport [10,23]. *Multi-flick* refers to the idea that flicks can potentially be used recurrently to easily explore a variety of scrolling distances. This is because a wide range of velocities can be specified, with very little change in position or movement, simply by adjusting the way in which consecutive flicks are combined. Han (TED Talk, www.ted.com) demonstrated the fluidity of multi-flicking off-screen content into and out of a multi-touch display. Hinrichs et al. [14] utilize multi-flicking with a stylus to control the flow of documents moving on the periphery of a tabletop. iPhone™ and iPod™ multi-flick as a natural form of scrolling through a list of videos and music.

These appealing manifestations of flicking reveal that designers are intuitively considering it to be a very natural form of interaction. However, very little knowledge exists about the effectiveness of multi-flick for navigating or scrolling through lists and documents. Flicking has been investigated in only one scenario; Reetz et al. [23] show that flicking objects with a stylus is an effective alternative to some of the currently available techniques for passing objects around a tabletop. However, their study does not explore the design space of multi-flick as a method for scrolling documents. As a result, we primarily question the efficiency of multi-flick for the common task of shifting the viewport when scrolling lists or documents.

We start by describing the design framework for multi-flick. To investigate the benefits and limitations of scroll-

ing using multi-flick, we then present a number of design variations of flick-based scrolling, including the commercial alternative available on the iPhone™. In a first study, we compare three variants of multi-flick with the standard scrollbar. Our results show that compound-multi-flick, a multi-flick variation we tested, was as good as the scrollbar for large tables and significantly better than the scrollbar in a tabletPC. Users also consistently preferred compound-multi-flick that combines flick-based scrolling with displacement-based scrolling to provide high-speed scrolling and fine control. In a second experiment we evaluated the various multi-flick designs for scrolling text documents. We found that for short documents, multi-flick techniques were slightly faster than the traditional scrollbar. Furthermore, the compound-multi-flick was often most preferred by the users whereas the commercially popular variant of multi-flick was least preferred. Our results offer several insights into designing with flick for the next generation of scrolling techniques in pen-based environments.

This paper makes the following contributions: 1) expands the design space of scrolling techniques to include multi-flick; 2) presents a framework to explore various designs of multi-flick; 3) through two experiments shows that flicking is as good an alternative as most current scrolling techniques and that users tend to view flicking as a preferred interaction technique for scrolling; 4) identifies defects with currently available multi-flick techniques.

### RELATED WORK
We review the related research on device-independent scrolling, touch- and pen-based scrolling, and flicking techniques that have been proposed, albeit without any empirical support for its role as a scrolling technique.

### Device-independent Scrolling
The scrollbar is the most common interface for navigating documents. Scrollbars are placed horizontally or vertically to the side of a document and are considered to be a zero-order positioning widget by which the user can shift portions of the document relative to the displacement of the scrollbar thumb. Shifting the viewport can become tedious particularly in 2D graphical workspaces [16]. Researchers [15,33] have also noted that operating a scrollbar consumes additional cognitive and motor resources, as the user has to shift their attention away from their primary task.

The most common alternatives, to overcome the limitations of scrollbars, appear as rate-based systems that are accessible using a mouse-wheel or IBM's ScrollPoint™. However, rate-based systems are limited by the amount of information that users can process visually [2], as fast scrolling rates cause disorientation. Researchers have studied a number of techniques to overcome the visual artifacts produced by rate-based systems. Speed-dependent automatic zooming (SDAZ) [15] integrates rate-based scrolling with automatic zooming. The document automatically zooms out when the scrolling rate increases. SDAZ gives users an impression that the document is scrolling at a constant rate which leads to less disorientation with large documents.

DDAZ (displacement-dependent automatic zooming) [7] scrolls and zooms in proportion to the amount of cursor displacement. DDAZ was faster for scrolling than SDAZ and other variations based on models proposed by van Wijk's and Nuij's model [27]. In Flipper [26], a variation that utilizes SDAZ, users can scroll at high rates one page at a time. Results show that users are faster with Flipper than with SDAZ. In a further development, Cockburn et al. [6] designed space-filling thumbnails (SFT) that provides an overview of all the pages with thumbnails and detail contents of the thumbnails by hovering over them. Results from that study show SFT outperforms all variations of SDAZ, including Flipper. However, for SFT to work well, the document needs to be separated into logical units.

### Pen- and Touch-based Scrolling
To a large extent, and with the exception of CrossY [1], scrolling systems for pen-based devices are primarily based on variations or virtual implementations of the scroll ring [29]. A scroll ring is designed as a doughnut-shaped touch-pad that scrolls a document in response to a user's circular strokes. A rotating scroll wheel that is prevalent on devices such as the iPod™ has seen widespread success. In one study users were more effective with the scroll ring than the mouse wheel [29]. The results also suggested higher performance gains with the scroll ring for larger scrolling distances. One explanation is that users scroll with one continuous movement with the scroll ring in comparison to multiple smaller strokes with the mouse wheel.

Inspired by the touchpad scroll ring [29], Moscovich and Hughes [22] designed a virtual scroll ring (VSR). The VSR mapped size of the scroll ring and movement speed to the scrolling rate of the document. Large or fast circular movements produced rapid scrolling, while small or slow movements yield slower document scrolling. Their results show that the VSR is comparable to the mouse-wheel and heightens users' experience in scrolling large documents.

The radial scroll tool [25], designed for touch displays, provides a scrolling widget that consists of a circle divided into segments. As the user crosses each segment, the document scrolls a pre-defined amount. Even though the scroll tool requires the user to maintain visual focus on the scroll widget it was more effective than a scrollbar for short scrolling distances. Curve dial [24] was designed to overcome the limitations of the radial scroll tool by using the size of the circle drawn to control scrolling speed; larger circles result in slower scrolling rates. Curve dial is appealing as the user can scroll the document in one continuous stroke drawn anywhere on the device surface. However, it requires constant contact with the surface of the display, which can occlude content being shifted into the viewport.

### Flicking Mechanisms
Flicking is analogous to a throwing motion in the real world which has also been the basis of several interaction

techniques. According to theories of naïve physics, the human perceptual and cognitive system simplifies occurrences of physical events; this simplification can lead to erroneous judgments about the estimated distance and trajectory traveled by an object, or of its velocity during movement [28]. Despite these limitations, researchers have proposed flicking as a compelling interaction technique for navigating and throwing objects in a virtual environment.

Geiβler's throw technique [10] requires the user to make a short stroke over a document, in the direction opposite of the intended target, followed by a long stroke in the direction of the target. The length of the short stroke determines the distance the document travels. Similarly, Wu et al. [32] describe a 'flick and catch' technique, in which an object is 'thrown' once it is dragged up to a certain speed. Kruger et al. [20] extend a rotation and translation technique to include a flicking action for passing and moving items on a tabletop. Hinrichs et al. [14] utilize flicking with a stylus to control the flow of a stream of documents moving on the periphery of a tabletop. Their system maps the length of the flick gesture to the speed of the document. However, none of these asses the benefits of flicking in a controlled evaluation as it was only featured as a secondary mechanism.

A paper by Reetz et al. [23] was one of the first to demonstrate the benefits of flicking as a method for passing documents over large surfaces. Flicking was designed to mimic the action of sliding documents over a table, and closely resembles the push-and-throw model designed by Hascoet [12]. In a study that required users to flick objects in target locations, flicking was found to be much faster than the radar technique for tabletop systems [23]. The authors concluded that throwing-based techniques show promise for improving interactions with tabletops.

Flicking also appears in different instances as a navigation technique. Johnson [17] proposed a panning interface on a touch-controlled display, which consisted of panning by pushing on the background with a simple flick movement. The document moved in the direction in which the user swiped their finger. Irani et al. [16] implemented a panning system which shifted a document at a distance and speed proportional to the displacement of the flicked cursor.

When using flick to access off-screen content, users usually employ multiple repeated flicks to get to the off-screen content. Jeff Han (TED talks) and video snippets showcasing iPhone™ demonstrate flicking (and consequently multi-flicking) as a natural form of accessing off-screen content.

## DESIGN FRAMEWORK FOR MULTI-FLICK

We propose a design framework to identify various factors that influence the performance of multi-flick scrolling. While we isolate two major factors, the mapping function and visual feedback that designers can control in the design of multi-flick, there exist several other elements, we refer to as external factors that designers often cannot control.

### External Factors

External factors that can affect scrolling performance with multi-flick are target distance, target type, user familiarity with the document, input resolution and real-estate for input. Ideally, any new scrolling technique should be minimally influenced by target distance. However, numerous studies have shown that scrolling mechanisms are affected by various document sizes and target distances [2,6,13,15]. Additionally the perceptual characteristics of a target can also influence scrolling performance [7]. Text and unformatted targets are harder to locate than formatted text or graphical images. As a result, most studies on scrolling [2,15] present the targets as salient items so that limitations of the human visual search processes do not confound with the performance of the scrolling techniques. The level of familiarity a user has with a document has a direct effect on scrolling performance, i.e. users take less time to find an object if its position in the document is known [13,2]. The spatial or temporal resolution of the input device should provide a sufficiently rich stream of input data to accurately model a single flick action so that it feels fluid and natural as in the real-world. Finally, the amount of space the user has to perform the flick can greatly influence their perception of the fluidity and naturalness of the action. On small devices, users might find it much harder to impart a natural flick than on a large table. Multi-flick is not immune to any of these external factors and designers need to be wary of these.

### Visual Feedback

With scrolling tasks, visual feedback typically consists of a smooth or a blurred transition from one part of the document to another. To reduce the effects of blurring, researchers have resorted to zooming out at high scrolling speeds [15]. Zooming with multi-flick needs to be carefully assessed as it is difficult to perceive the document's contents at varying magnification levels. An alternative to zooming the document is to provide transient visual cues that hint at the content of what will appear next in the scrolling sequence [18,30]. Transient visual cues can lead to occlusions which can be detrimental on devices with small view ports.

An alternative to zooming or using transient cues is to use simple animation models for multi-flick scrolling. Klein and Bederson [19] demonstrate that animating the movement of the document during the scrolling operation can improve target search tasks by up to 5.3% for text targets and 24% for graphical targets. Although animation can enhance scrolling performance, Andersen [2] suggests that scrolling rate should be limited by the maximum rate a target can be perceived at during the animation. Anderson's [2] recommendation can be extended to multi-flick scrolling by limiting the animation to the rates proposed in the literature [7,8,26,31]. These have shown to be about 2100 pixels/sec.

## Mapping Function

Scrolling mechanisms are driven by a mapping function that performs a translation of the user's manipulation of the input device to the scrolling operation. The mapping function primarily affects scrolling speed. In a zero-order position mapping, like in the classical scrollbar, the relative displacement of the cursor produces a proportional scrolling distance. With a first-order rate mapping, such as rate-based scrolling, the mapping function translates the displacement of the cursor (or some other value such as force applied on an isometric input device) to a scrolling speed. Longer or shorter displacements of the cursor from the marker cause the document to scroll at a higher or lower rate, respectively. Studies suggest that either zero-order or first-order mappings operate best based on the context of the user's task [13]. However, most novel variations of scrolling techniques that outperform traditional scrollbars are designed on the premise of a first-order rate-based mapping. Additionally, the mapping function needs to take into consideration the effect that target distance may have on scrolling performance. Studies show crossover effects between scrolling mechanisms and target distance [15,29]. For example, short but frequent displacements, as produced with a scroll wheel are less efficient for scrolling large documents than one continuous and fluid movement [29].

Multi-flick is based on consecutive flick actions. With a single flick the rate of displacement of the pen-tip maps to scrolling speed. Generally, in multi-flick, each successive flick maps the rate of displacement plus the current scrolling speed to a new scrolling speed. This allows Multi-flick to employ various types of mapping functions to control scrolling speeds. The mapping function needs to strike a balance between the frequency of the flicks and the movement speed. While such a design constraint cannot be computed in advance, multi-flick designs should minimize the number of strokes required to scroll a document.

## DESIGN AND IMPLEMENTATION OF MULTI-FLICK

The main focus of our design was to explore the effectiveness of different mapping functions. The various multi-flick techniques we propose explicitly address the challenge of intuitively mapping elements of successive *input flick-gestures* onto the *document scrolling speed*. We describe the basic flick before presenting the multi-flick variants.

## Input Device and Flick

In the process of flicking with a pen the user imparts speed and direction to the object in one quick motion. Such an action can be decomposed into an initial and final velocity, an acceleration value, and a direction vector that defines the trajectory of the object. All the multi-flick designs we evaluated are based on the velocity input model described by Reetz et al [23], i.e. to scroll a document, the user puts the pen down on the document, drags the pen towards the desired direction and then releases the pen to scroll the document. The document will scroll forward and scroll

backward, if the directional vector appears in the top-right quadrant or the bottom-left quadrant of the display, respectively. This fluid action assigns to the interaction an initial velocity and direction (see Figure 1). To calculate the pen velocity and direction we maintain a history of the last ten velocity-direction samples. When the user lifts the pen, we apply a linear regression to the collected data to estimate the liftoff speed and direction imparted to the document.

## Mapping Functions

We can map consecutive flick actions to the document

scrolling speed in numerous ways - here we explore four mappings by manipulating the current document speed.

### Multi-flick-add (MFA)

In MFA, scrolling speed is manipulated using the rate of the flicking action as well as its direction. Low or high flicking rates continuously decrease or increase the scrolling speed, respectively. The direction of the flicking action can increase or decrease the scrolling speed. As a result, successive flicks result in adding or subtracting (depending on the direction of the flick) from the current document scrolling speed. Figure 2, shows the relationship between the document speed to pen-flicking speed. To stop scrolling the user taps the stylus on the screen. The relationship between flicking rate and document scrolling rate is as follows:

$$V_S = V_{CS} + m \times (d \times V_{FL}),$$

where $V_S$ – new scrolling speed, $V_{CS}$ – current scrolling speed, $V_{FL}$ – speed of the flick movement, m – C/D ratio, d is ±1 based on the direction of the flick. When $V_S$ is negative, the document starts scrolling in the reverse direction.

### Multi-flick-standard (MFS)

MFS is the default multi-flick technique that would become available to the user if a flick is supported by the system. MFS directly maps the flick speed to the scrolling speed:

$$V_S = m \times V_{FL}$$

Figure 2b shows the relationship between the document speed and pen-flicking speed. There are several differences between MFA and MFS. In MFA to reach a higher speed, the user needs to invoke a few consecutive flicks, whereas with MFS the user can achieve this with one fast flick movement. MFA and MFS also differs with respect to the mechanism used for reversing the scrolling direction. In MFA the user needs to flick several times in the opposite direction to scroll in the reverse direction or can tap to stop scrolling and then flick in the opposite direction. With MFS one flick movement in the opposite direction is sufficient to reverse the scrolling direction of the document.

### Multi- flick-friction (MFF)

We implemented MFF using the flick implementation employed in the iPhone™. MFF is similar to MFS but includes an additional friction "factor" that gracefully reduces the document scrolling speed after some time interval. In essence, MFF is the default multi-flick action that would be available in a system that implements flick with a friction. The "friction" coefficient determines how long it

takes the document to reduce speed to the point of stopping. The relationship between flick rate and document rate is

$$V_S = m \times V_{FL} - \mu t,$$

Where μ is a friction coefficient and t is elapsed time from flick lift-off (Figure 2c). In comparison with previous variations, MFF is the only multi-flick technique that does not require any explicit actions to stop scrolling.

### Compound-multi-flick (CMF)

MFA, MFR and MFF do not provide any feedback to the flick action until the user lifts the pen from the surface. Lack of visual feedback can be disorienting to users. Additionally, in all of the three previous mappings, users performed a complete flick action, i.e., stroke+lift-off. Since scrolling speed is unknown to the user, completing a flick before seeing the document scroll can lead to over/undershoot in tasks that require searching for a specific part of the document. To overcome these potential
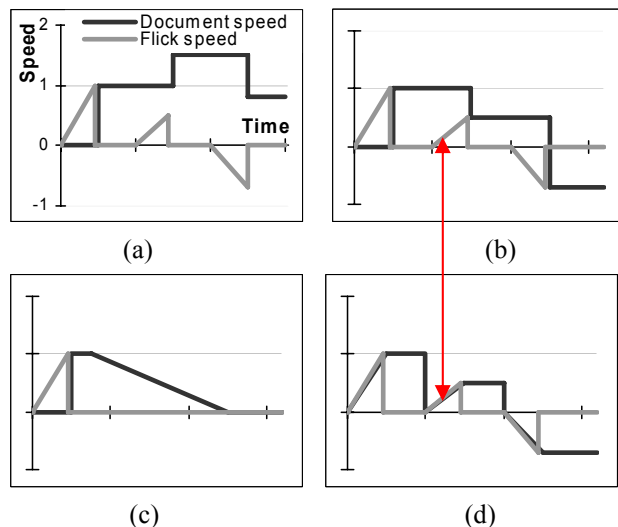


Figure 2: The mapping functions for the different multi-flick techniques with mapping coefficient m equal to one. (a) MFA, (b) MFS, (c) MFF, (d) CMF.

limitations and to provide users with greater control in scrolling, we propose CMF. As the name suggests, CMF is a compound technique that combines flicking with a displacement-based control. When the user glides the pen over the surface, the document displaces by a distance equivalent to the displacement of the pen (see Figure 2d). When the user lifts the pen from the surface with a quick thrust a flick is registered and the flick speed directly maps to the scrolling speed. The users can stop the document from scrolling by putting the pen on the document. Users can also have fine control over scrolling by gliding the pen on the surface.

As a result this multi-flick variation operates in two states each of which influences a different variable in the scrolling operation: the displacement distance of the document and the scrolling speed. As the pen is on the surface and being adjusted the following relation is upheld: $DIS_D = DIS_P$, where $DIS_D$ is the document displacement and

$DIS_P$ is the pen displacement. As the pen is released, the relationship with document scrolling speed is $VS = m \times VFL$.

Additionally to allow fine control over document displacement (for precise positioning) without initiating a flick, we introduce two additional parameters: timeout and minimal liftoff speed. If, while doing a pen movement, the pen touches the display surface for a time shorter than the timeout period, the movement is interpreted as a flick and the document scrolls accordingly. If the pen-movement time is longer than the timeout, the liftoff speed is checked against the minimal liftoff speed. The document scrolls if the flick liftoff speed is larger than the minimal liftoff speed.

### Visual Feedback

The multi-flick visual feedback consists of simply animating the scrolling of the document. The maximum document scrolling speed is set to a rate of 2100 px/second (~3 pages) based on the results of previous research [2,20]. While several other types of feedback are also possible, we did not want to confound our results with different variations of visual feedback in this preliminary investigation of multi-flick mappings. Furthermore, common and widespread scrolling mechanisms do not rely on zooming or transient visual cues but instead consist of simple scrolling motion.

### Fine Tuning Parameters

In several preliminary studies we identified the ideal parameters for each of the various flicking techniques. Besides identifying the parameters we also observed that MFA was on average slower and the least preferred technique by the participants. Based on these results MFA was dropped from further investigation. The mapping coefficient m was set at 0.5 for document window sizes larger 120 x150 mm to allow users to reach a maximum speed with a one very fast movement. For smaller window sizes we found that at $m = 0.5$ the document scrolled too fast for finer control so m was set at 0.2. The timeout for CMF was set at 0.5s, and threshold liftoff speed 1000 px/s. For MFF μ was set to 750 px/s$^2$.

### STUDY: A COMPARISON OF MULTI-FLICK DESIGNS

The main goal of this study was to compare the various multi-flick mapping functions. Three MF techniques were tested in the experiments MFF, MFS and CMF. MFF was included since it's a default technique used in commercial applications (iPhone™), and the only technique that does not require an explicit action to stop scrolling. MFS was included as a baseline MF technique that becomes available to the user if a system supports flick. Additionally we contrast their performance with a scrollbar, which is a standard scrolling technique that is readily available on pen based systems. Limited by the number of tasks that can be tested we only considered visual search in unknown documents.

As reported in [2,7,15,19], users employ different strategies in scrolling documents of various types. We therefore,

conducted two separate experiments; in the first experiment we compare multi-flick designs on lists and in the second we evaluate our designs with text documents.

### Apparatus

To examine the effect of display size on multi-flick we carried out our experiment on three different device sizes: standard TableTop – 590×440 mm (document window size 350×425 mm), tabletPC - 210×160 mm (document window size 120×150 mm), and PDA – 78×58 mm (document window size 78×58). For a TableTop we used a top-projected display (1024×768) with an A2 Wacom digital board. The system ran on Intel P4 machine. For the PDA and tabletPC the experiment ran on a 1.20 GHz tablet with 512 MB RAM and 1024x768 display resolution.

We set the active area for viewing the list on the table and tabletPC to 575×722 pixels and on the PDA to 282×376 pixels. We developed the experimental setup using C#. All techniques were carried out with a digital pen that came with the tabletPC. We used the same tabletPC to simulate the PDA configuration by placing a cardboard cutout to restrict the display and interaction area to a typical PDA size. The video figure submitted with the paper shows the experimental system and demonstrates the techniques.
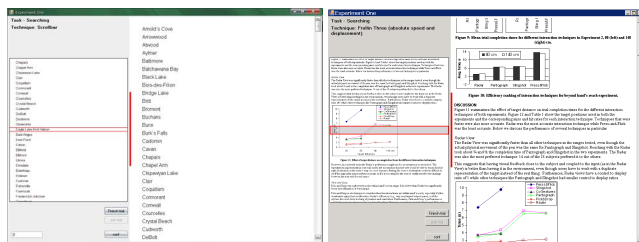


**Figure 3: The screen view of the experimental task for lists (left) and documents (right).**

The experimental software recorded trial completion time (MT); and number of crossings (NC) as dependent variables. MT is the total time taken to complete the task and is defined as the time taken for the user to scroll to the correct visual item. The counter begins at the moment the user clicks on the "start-trial" button and stops when the user clicks the "finish" button. Users were not allowed to continue to the next trial without successfully completing a trial. NC is the number of times the target line (highlighted on the left pane) crosses the red tolerance bar. In an exit questionnaire we asked participants to rank the different scrolling functions in order of preference.

### Experiment 1: Comparing Multi-flicking on Lists

#### Task and Stimuli

Figure 3 (left) shows the experimental setup. Each task began with the display of a new target list item on the left part of the screen and the document window on the right. The list window had a size of 575×722pixels.

The task consisted of locating a target city on the left part of the screen from an alphabetical list of Canadian cities and positioning it on the right part of the screen so it is

positioned within the area indicated by the red vertical line at the center of the screen (see Figure 3). This line allowed a maximum displacement error of ±150 pixels.

### Procedure and Design

We carried out the experiment with 8 participants (4 females and 4 males, 7 – right-handed) between the ages of 25 and 32. All subjects were tested individually. The experiment used a 3×4×2 within-factor design with a variety of planned comparisons. The independent variables were *Device size (table, tabletPC, PDA)*, *Multi-flick Techniques* (MFS, MFF, MFP and Scrollbar), and *list length* (short - 100 items, and long - 400 items). The order of presentation of the different techniques and devices was balanced using a (partially-balanced) Latin-square design. For short lists the target item was always one of items 20, 50 or 70, while for long lists it was always one of items 200, 280 or 360. The order of target and lists were randomized. Furthermore, to avoid learning effects each trial used a unique list of randomly selected and alphabetically ordered Canadian cities.

For each technique and device, participants completed 1 training trial and 12 test trials (2 for each target item), for a total of 12 training trials and 144 test trials. The participants on average took 60 minutes to complete the experiment.

### Results

We used the univariate ANOVA test and Tamhane post-hoc pair-wise tests (unequal variances) for all our analyses.
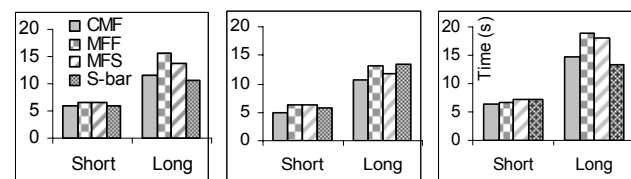


**Figure 4: Mean trial completion times of each technique and list length for the three devices – Large Table (left), TabletPC (Center) and PDA (Right). Legend is the same for each graph.**

*Movement Time:* The average trial completion time was 10.03s with a standard deviation (s.d.) of 5.71s. There was a significant interaction of device, list length and technique on MT ($F_{(6,42)}$ = 4.93, p<0.01). Subsequent analysis showed that there was a significant interaction of list length and technique on MT for the large table ($F_{(3,21)}$ = 5.18, p<0.01) and the PDA ($F_{(3,21)}$ = 11.65, p<0.001) but we found no interaction for the tabletPC ($F_{(3,21)}$ = 2.85, p = 0.062). Figure 4 shows the mean MT for each device and list length.

For the table, at short target distances, we found no difference in MT between the techniques ($F_{(3,21)}$ = 1.331, p=0.291). The fastest technique was CMF followed by the Scrollbar, MFF and MFS. For large distances there was a significant difference between the techniques ($F_{(3,21)}$ = 8.37, p<0.001). The fastest technique was Scrollbar followed by CMF, MFS and MFF. Pair-wise comparisons showed significant difference for all pairs except (Scrollbar, CMF).

For the tabletPC, there was a significant difference between techniques ($F_{(3,21)}$ = 5.313, p<0.01) and lists ($F_{(1,6)}$ = 100.01, p<0.01). The fastest technique was CMF followed by MFS, MFF and Scrollbar. The following pairs were significantly different (CMF, MFS), (CMF, MFF), and (CMF, Scroll-bar). Users were significantly faster on short than long lists.

For the PDA, at short distances we found no significant difference between the techniques ($F_{(3,21)}$ = 0.446, p=0.723). The fastest technique was CMF followed by MFF, MFS and Scrollbar. At large distances there was a significant difference between the techniques ($F_{(3,21)}$ = 11.6, p<0.001). Scrollbar was the fastest technique followed by CMF, MFS and MFF. All pairs were significantly different except the following (Scrollbar, CMF) and (MFS, MFF).

*Number of Crossings:* Overall, we found a significant difference in NC between the techniques ($F_{(3,21)}$ = 12.771, p<0.01) and device ($F_{(2,14)}$ = 6.95, p<0.001) but not for document length ($F_{(1,7)}$ = 0.305, p=0.58). We found no significant interaction between the factors. Pair-wise comparison of the techniques showed that Scrollbar had significantly fewer NC than all other techniques, followed by MFF, MFS and CMF. We found no significant difference between the various multi-flick techniques. There were significantly fewer crossings with a large table than with a tabletPC or PDA.
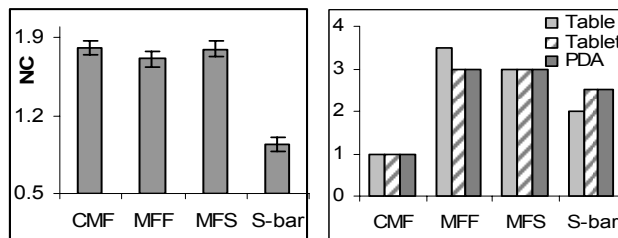


**Figure 5: (Left) Average number of crossings for each technique; (Right) Median user preferences across devices. User preference with 1 =most-preferred, 4 = least preferred.**

*Subjective Preference:* Overall users preferred CMF the most and MFF the least. The rank order of the techniques was the same for all devices with CMF preferred most, followed by scrollbar, MFS and MFF. Figure 5b shows the median user preference for each device.

### Discussion

The results of our study show that CMF is the fastest and most preferred multi-flick technique. Furthermore, CMF is at least as good as the scrollbar for both small and large lists, across all display conditions. We explain our findings by looking more closely at the detailed log files and our observations during the experiment.

*Time to Initiate the Interaction:* To perform a trial, participants were required to first click on the 'Start Trial' button, locate the target, and then finish the task by clicking the 'Finish Trial' button. Our log files showed that users were significantly faster initiating multi-flick than acquiring the scrollbar. Users took on average 2.6s to initiate the task

with the scrollbar in comparison to 1.4s with CMF and 1.8s with MFS and MFF. The overhead in initiating a multi-flick is lower than with scrollbar as it was possible to apply the flick gesture anywhere over the entire document. However, users performed a high index-of-difficulty target acquisition task to acquire the scrollbar. On tabletPCs and PDAs the parallax errors created by the physical space between the pen's contact surface and the actual display adds the problems of controlling the small scrolling widgets.

*Precision vs. Number of Crossings:* The average difference between actual position of the target item and the position at which the document was scrolled to, showed that with CMF and scrollbar participants were more precise, than with other multi-flick techniques. More detailed analysis of recorded pen movements revealed that most of the crossings for multi-flick techniques occurred at the final part of the trial where the targeted item was already visible on the screen. Furthermore, the size of the overshoots was generally smaller for CMF than for MFF and MFS. This indicates that with CMF participants tended to scroll list items closer to the center of the allowed range, while with MFF and MFS they would usually finish the trial immediately when the item was within the acceptable range.

*Handedness:* We observed that the one left-handed subject had to reach over to the right-side to acquire the scrollbar, which led to some frustration and difficulty. Their interaction suffered from occlusion that takes places when manipulating the scroll bar with the left hand. However, the same participant had no difficulty in using the multi-flick technique and indicated much greater satisfaction with the multi-flick techniques over the scrollbar. We recognize that multi-flick is unaffected by handedness and feel that this property is extremely appealing.

### Experiment 2: Comparing Multi-flick for Documents
Here we used the same apparatus as in experiment 1.

*Task and Stimuli*
Figure 3 (right) shows the experimental setup. Each task began with the display of a new target page on the left part of the screen and the document window on the right. The target page preview was large enough for users to legibly see the body of the text (380×460 pixels). Page numbers were removed from the page images. The document window had a size of 575×722pixels.

The task consisted of locating and positioning on the right pane the target page and highlighted text presented on the left pane. To complete the trial, the user should position the highlighted text within the red vertical line at the center of the screen (Figure 3). The red line indicated the maximum permissible tolerance of ±150 pixels for displacement error.

*Procedure and Design:*
The experiment used 8 subjects (4 males and 4 females, 7-right-handed, between ages 25 to 33). None of these subjects participated in experiment 1 and all were tested indi-

vidually. The experiment used a 2×4×2 within-factor design with a variety of planned comparisons. The independent variables were d*evice size* (table and tablet), *scrolling techniques* (MFS, MFF, MFP and Scrollbar), and *document length* (short with 20 pages and long with 100 pages). For short documents targets were always at pages 5, 11 or 18, while for large documents the targets were at pages 30, 50, or 70. To complete the experiment in 2 hrs we excluded the PDA device. The order of presentation of the different techniques and devices was balanced using a Latin-square design. The target order was randomized.

To avoid learning effects, 16 unique HCI documents containing one picture per two pages, were used for the 3 short and long distances. Each document was randomly assigned to a trial. Participants completed 1 training and 12 test trials (2/document length) per technique and device, for a total of 8 training trials and 96 test trials and took about 100 mins.

### *Results*
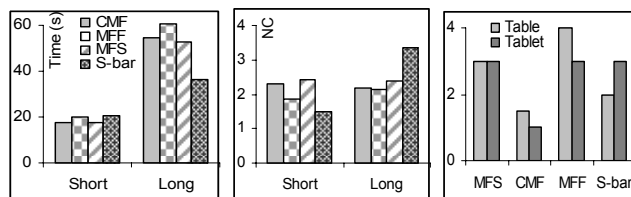We used the univariate ANOVA test and Tamhane post-hoc comparisons (unequal variances) for all our analyses.



**Figure 6: Comparison of the interaction techniques in terms of (a) Mean completion time, (b) number of crossings and (c) User preference with 1 =most-preferred, 4 = least preferred.**

*Movement Time:* The average MT was 35.33s, s.d=32.05. There was a significant interaction between the techniques and document length ($F_{(3,22)}$ = 14.25, p<0.001) and we found no significant difference between devices ($F_{(1,7)}$ = 2.93, p = 0.13). Figure 6-Left shows the mean trial completion time for each technique and target distance. For short documents, we found no significant difference between the techniques. The fastest technique was CMF followed by MFS, MFF and Scrollbar. For long documents there was a significant different between the techniques ($F_{(3,21)}$ = 9.69, p < 0.001). The scrollbar was significantly faster than all other techniques, followed by MFS, CMF and MFF. We found no significant difference between multi-flick techniques.

*Number of Crossings:* The average NC was 2.25, s.d.=1.926. Overall, we found a significant interaction in NC between document length and techniques ($F_{(3,21)}$ = 5.83, p<0.01). We found no significant difference between devices ($F_{(1,7)}$ = 0.63, p=0.46). Figure 6(b) shows the mean NC (technique/ distance). For short documents we found a significant difference between the techniques ($F_{(3,24)}$ = 8.16, p < 0.001) with the scrollbar having least NC, followed by MFF, CMF and MFS. There was a significant difference between the following pairs (scrollbar, CMF), (scrollbar, MFS), and (MFF, MFS). For long documents we found no

significant difference between the techniques ($F_{(3, 24)}$ = 2.62, p = 0.073). CMF had the fewest NC followed by MFF, MFS and Scrollbar.

*Subjective Preference:* Overall users most preferred CMF and least preferred MFF for both display sizes (Figure 6(c)). The rank-order of the techniques for both devices was the same. However for large tables, the scrollbar was a close second to CMF whereas for the tabletPC both scrollbar and MFS were equally preferred for the 2nd spot.

### Discussion
For short documents user performance with multi-flick techniques is at least as good as their performance with the scrollbar. However, for long documents the scrollbar was significantly faster despite having higher target crossings. Overall, CMF was most preferred and MFF the least.

We examined our techniques for each target distance (5, 11, 18, 30, 50 and 70 pages) and found that for target distances of 30 and 50 pages the scrollbar was significantly faster than the multi-flick techniques, but for targets at 70 pages the scrollbar was marginally faster than multi-flick (i.e, we found no significant difference between Scrollbar and CMF). The results suggest that relative to the scrollbar the multi-flick techniques dipped in performance for target distances between 30 and 70 pages but for distances larger than 70 pages we expect that they would perform better.

The scrollbar performs best in the mid-range documents for several reasons. First, the amount of overhead in acquiring the scrollbar is significantly lower when the user has to scroll beyond the short range distances. We see that an average scrollbar movement in the medium range targets varies between 20 and 35s, resulting in an 8% overhead in medium ranges in comparison to a 14% overhead in shorter ranges. Secondly, users can maintain their pen on the scrollbar and drag at a rate that is sufficiently fast and self-controlled. However, the difference between scrollbar and multi-flick diminishes again in the large target distances. Users complain of fatigue when they have to maintain pen contact with the surface beyond 40s to glide the pen across the scroll widget. While we did not test the techniques for targets beyond 70 pages we believe that user fatigue with the scrollbar sets-in when maintaining the pen pressure and glide the pen across the surface.

### GENERAL DISCUSSION
The results of both experiments show that different multi-flick designs can lead to significant performance differences for different platforms and document types. Here we discuss the findings of both the experiments and identify benefits and limitations of various multi-flick techniques.

### *Multi-flick*
Our results show that multi-flick is a promising technique that outperforms the common scrollbar under numerous conditions. With short range documents and documents organized into alphabetically ordered lists, users can comfortably control document scrolling rates with a multi-flick technique.

Our results suggest that tiny variations in multi-flick design can lead to differences in performance.

*Ease of Use:* Particularly appealing with Multi-flick is the amount of time users require to grasp the technique. In both experiments we gave users only one training trial after demonstrating the technique. We compared MT vs. Trail number for each device and document length. Our comparison showed that deviations from a horizontal line were between 0.3 and -0.5 degs. This suggests that one training trial was sufficient to obtain the performance rates we obtained in our experiment and Multi-flick can be considered a walk-up-and-use technique. Additionally, participants consistently commented on the ease of using a natural gesture based technique for scrolling documents with a pen.

*Motor and Cognitive Benefits of Multi-flick:* While our results suggest that Multi-flick is as good as scrollbars, our observations highlighted certain motor and cognitive benefits of Multi-flick that we had not identified prior to our evaluation. As mentioned earlier, Multi-flick is unaffected by handedness, i.e. left-handed or right-handed users can work with Multi-flick without having to reconfigure any settings. Additionally, with multi-flick users do not need to be actively and continuously focused on the scrolling interaction. This can reduce the level of cognitive effort required by users as their eyes can remain focused on the document instead of splitting their attention between the document and the mechanics of the interaction. Finally, for very long documents users complained about the length of the continuous operation required with the scrollbar. Multi-flick designs can simply have the user lift-off the pen after a stroke and can passively observe the flow of the document without having to initiate any interaction.

*Inking vs. gesturing:* In many cases multi-flick requires a mode switch. To avoid explicit mode switching the rapid flick gesture can be interpreted as an implicit mode switch.

### CMF
Our results show that CMF generally outperforms and is the most preferred of all the Multi-flick variations. We can attribute this to several factors: fine control over movement, reduced number of flicks, and constant document speed.

*Fine Document Control:* Fine control over document displacement and speed allowed users to stop the scrolling and then quickly bring the desired item (page) within the targeted range. We observed that this feature of CMF, more than other aspect of this technique reduces the number of over/undershoot operations in comparison to MFF and MFS. This observation led us to conclude that designers should include a level of fine-grain control over scrolling speed with any Multi-flick implementation.

*Constant Document Speed:* Unlike MFF and MFS, document scrolling speed remained constant in CMF. In MFF, document speed gradually decreased and in MFS users could increase and decrease document speed intermittently. The gradual decrease in document speed allowed users to assess where they are in the document, and decide to continue or stop scrolling. However, for long documents, variable document speed led to a large number of consecutive flicks, thereby affecting performance. While our results and observations do not allow us to prescribe whether Multi-flick designs should use constant or variable document speed, we believe that the implementation that will lead to the minimal amount of flicks should guide the design.

*Reduced Number of Flicks:* The number of flicks required to scroll a document grows with the target distance, the difficulty in locating a page (if there are not any easily recognizable cues), and to a large extent depends on the Multi-flick design. Based on our results, the number of flicks performed reveals that both CMF and MFS had fewer flicks than MFF (for lists on a tabletPC the average was about 3.1 for MFS, 3.25 for CMF and 6.8 for MFF). For some trials in the list study tasks were completed in a single flick.

### MFF
We initially expected MFF to be the most preferred technique as it resembles how we interact daily with objects. However, MFF and MFS were least preferred and performed worst overall. MFF and MFS were not significantly different from CMF and scrollbar on short distances. More than 60% of time was spent scrolling large documents. It is possible that user preference was based on a technique's ability to scroll large documents rather than short documents.

Most commercial systems use a Multi-flick similar to MFF. However designers should carefully consider what variation is best for a given application. The wrong choice could negatively influence users' perceptions of Multi-flick and garner a lack of appreciation for its naturalness.

### Lessons for Practitioners
The following recommendations emerge from our study:

- We believe that in pen or stylus systems, multi-flick scrolling is an effective alternative to conventional scrolling. Its natural form of interaction makes it appealing and conducive to widespread adoption by users.

- Designers need to consciously design for multi-flick as our study shows that the default implementations (in our case both MFS and MFF) that come with the design of a Multi-flick are not as good as the CMF technique.

- Multi-flick techniques scale well across display sizes and tasks and our experience suggests that a multi-flick technique without a friction parameter is preferred over commercially popular implementations.

- Other combinations of alternatives such as controlling magnification and scroll rates need to be considered.

### CONCLUSIONS
We identified various factors that influence scrolling in pen based systems and investigate the benefits and limitations of flick based solutions. We present the design for three multi-flick techniques and compare them to the standard scrollbar over three different display sizes for scrolling through an alphabetically ordered list and a text document. The findings of our study show that Compound Multi-flick (CMF) can be

as effective as using the scrollbar and is most preferred by users. Our results also highlight significant differences in preference and performance between various implementations of Multi-flick. Given that flicking is being deployed in commercial tools, it is not the improved performance with multi-flick that will make this technique easily accepted by novice users. Instead, users will adopt this technique and prefer it to its counterparts because of the ease and satisfaction it brings to the user's experience.

**ACKNOWLEDGEMENTS**

**REFERENCES**

1. Apitz, G. and Guimbretière, F. CrossY: a crossing-based drawing application. In Proc. UIST '04, 3-12.

2. Andersen, T. H. A simple movement time model for scrolling. In CHI'05 Extended Abstracts, 1180-1183.

3. Ayatsuka, Y., Rekimoto, J., and Matsuoka, S. Popup vernier: a tool for sub-pixel-pitch dragging with smooth mode transition. In Proc. UIST '98, 39-48.

4. Bartlett, J. F. 2000. Rock'n'Scroll Is Here to Stay, IEEE Computer Graphics and Applications, 20(3), 40-45.

5. Byrne, M. D., John, B. E., Wehrle, N. S., Crow, D. C. The tangled Web we wove: a taxonomy of WWW use. In Proc. CHI'99, 544-551.

6. Cockburn, A., Gutwin, C., Alexander, J. Faster Document Navigation with Space-Filling Thumbnails. In Proc CHI'06, 1-10.

7. Cockburn, A., Savage, J., and Wallace, A. Tuning and testing scrolling interfaces that automatically zoom. In Proc. CHI '05, 71-80.

8. Cooper, K., de Bruijn, O., Spence, R., Witkowski, M. A comparison of static and moving presentation modes for image collections. In Proc. AVI'06, 381-388.

9. Evans, K. B., Tanner, P. P., Wein, M. 1981. Tablet-based Valuators that Provide One, Two or Three Degrees of Freedom. Computer Graphics 15 (3), 91-97

10. Geißler, J. Shuffle, throw or take it! working efficiently with an interactive wall. In CHI'98 Extended Abstracts, 265-266.

11. Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C., and Want, R. Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In Proc. CHI'98, 17-24.

12. Hascoët, M. Throwing models for large displays, Proc. of British HCI'2003, 73-77.

13. Hinckley, K., Cutrell, E., Bathiche, S., Muss, T. Quantitative analysis of scrolling techniques. In Proc. CHI'02, 65-67.

14. Hinrichs, U., Carpendale, S., and Scott, S. D. Evaluating the effects of fluid interface components on tabletop collaboration. In Proc. AVI '06, 27-34.

15. Igarashi, T. and Hinckley, K. Speed-dependent automatic zooming for browsing large documents. In Proc. UIST '00, 139-148.

16. Irani, P., Gutwin, C., and Yang, X. D. Improving selection of off-screen targets with hopping. In Proc. CHI '06, 299-308.

17. Johnson, A. A comparison of user interfaces for panning on a touch-controlled display. In Proc. CHI'95, 218-225.

18. Kaptelinin, V., Mäntylä, T., and Åström, J. Transient visual cues for scrolling: an empirical study. In CHI '02 Extended Abstracts, 620-621.

19. Klein, C. and Bederson, B. B. Benefits of animated scrolling. In CHI '05 Extended Abstracts, 1965-1968.

20. Kruger, R., Carpendale, S., Scott, S. D., and Tang, A. Fluid integration of rotation and translation. In Proc. CHI '05, 601-610.

21. MacKenzie, I. and Riddersma, S. Effects of Output Display and Control-Display Gain on Human Performance. In Interactive Systems. Behaviour and Information Technology, 13 (5), 328-337.

22. Moscovich, T., Hughes, J. F. Navigating Documents with the Virtual Scroll Ring. In Proc. UIST'04, 57-60.

23. Reetz, A., Gutwin, C., Stach, T., Nacenta, M., and Subramanian, S. Superflick: a natural and efficient technique for long-distance object placement on digital tables. In Proc. Graphics Interface'06, 163-170.

24. Schraefel, M., Smith, G. M., Baudisch, P. Curve Dial: Eyes-free Parameter Entry for GUIs. In Proc. CHI'05 Extended Abstracts, pp. 1146-1147.

25. Smith, G. M. and Schraefel, M. C. The radial scroll tool: scrolling support for stylus- or touch-based document navigation. In Proc. UIST'04, 53-56.

26. Sun, L., Guimbretière, F. Flipper: a new method of digital document navigation, In CHI '05 Extended Abstracts, 2001-2004.

27. van Wijk, J. and Nuij, W. 2004. A Model for Smooth Viewing and Navigation of Large 2D Information Spaces. IEEE Trans. on Vis. and Comp. Graphics, 10 (4), 447-458.

28. Weld, D., and de Kleer, J. 1989. Readings in Qualitative Reasoning about Physical Systems, Morgan Kaufmann.

29. Wherry, E. Scroll Ring Performance Evaluation. In CHI'03 Extended Abstracts, 758-759.

30. Wirth, M. and Zhai, S. 2000. Visual scrolling feedback and method of achieving the same. US patent 6,476,83.

31. Wittenburg, K., Forlines, C., Lanning, T., Esenther, A., Harada, S., and Miyachi, T. Rapid serial visual presentation techniques for consumer digital video devices. In Proc. UIST '03, 115-124.

32. Wu, M. and Balakrishnan, R. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In Proc. UIST '03, 193-202.

33. Zhai, S., Smith, B. A., and Selker, T. Improving Browsing Performance: A study of four input devices for scrolling and pointing tasks. In Proc. Interact'97, 286-293.