Design and Evaluation of Finger-Count Interaction: Combining multitouch gestures and menus

Gilles Bailly[1], Jörg Müller[1], Eric Lecolinet[2]
[1]Quality and Usability Lab, Telekom Innovation Laboratories, TU Berlin, Germany
[2]Telecom ParisTech, CNRS LTCI UMR 5141, France.

Selecting commands on multi-touch displays is still a challenging problem. While a number of gestural vocabularies have been proposed, these are generally restricted to one or two fingers or can be difficult to learn. We introduce Finger-Count gestures, a coherent set of multi-finger and two-handed gestures. Finger-Count gestures are simple, robust, expressive and fast to perform. In order to make these gestures self-revealing and easy to learn, we propose the Finger-Count menu, a menu technique and teaching method for implicitly learning Finger-Count gestures. We discuss the properties, advantages and limitations of Finger-Count interaction from the gesture and menu technique perspectives as well as its integration into three applications. We present alternative designs to increase the number of commands and to enable multi-user scenarios. Following a study which shows that Finger-Count is as easy to learn as radial strokes, we report the results of an evaluation investigating which gestures are easier to learn and which finger chords people prefer. Finally, we present Finger-Count for in-the-air gestures. Thereby, the same gesture set can be used from a distance as well as when touching the surface.

1. INTRODUCTION

Multi-touch technologies opened a novel opportunity to increase the input bandwidth between the user and the machine. Thanks to two-handed and multi-finger interaction users can manipulate multiple degrees of freedom in a simple and coherent way. For instance, the introduction of the iPhone and the Microsoft Surface made it common to use two fingers to fluidly move, rotate and zoom a virtual object. While a large number of multi-touch technologies [Dietz 2001; Han 2005] and devices (Microsoft Surface, Apple iPhone, etc.) have been proposed both in academia and industry, there are only few studies focusing on:

- Coherent multi-touch gesture sets (especially with more than two fingers)
- Interaction techniques promoting autonomous learning of multi-touch gesture sets.

We present a novel coherent set of two-handed multi-finger gestures called Finger-Count gestures (Figure 1 and 2). These gestures are simple to understand and to perform, robust (supported by most multi-touch technologies), expressive (25 different gestures are available) and fast to perform. Finger-Count gestures exploit the natural ability of humans to count with fingers. Users only need to put *N* fingertips in contact with the interactive surface, the system just having to count the number of finger contacts. Twenty-five (5x5) different input configurations can thus be expressed when using both hands.

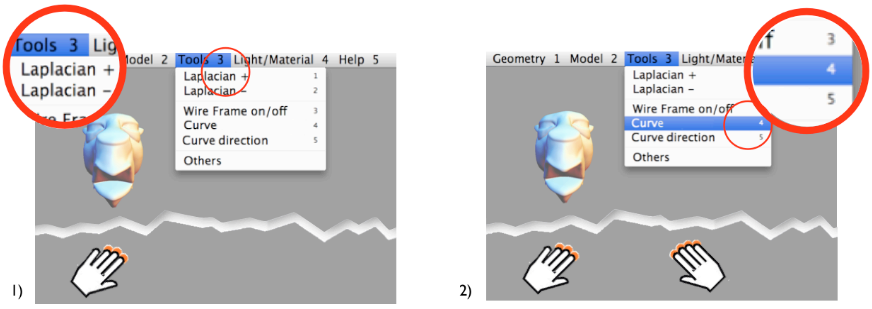
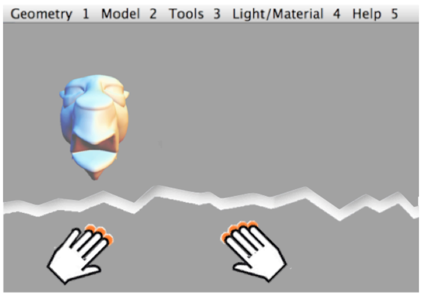


Fig. 1. Finger-Count in novice mode: The user executes the command "Curve" by executing the 4th command of the 3rd menu. (1) The user presses 3 fingers with the non-dominant hand (in this case, the left hand) to select the menu. The corresponding menu (which number appears next to its name in the menu bar) is displayed after 300ms. (2) The user presses 4 fingers with the dominant hand (in this case, the right hand) to select the item. The corresponding command (which number appears next to its name in the menu) is executed when the user releases all fingers. Numbers hence act as menu shortcuts. As for keyboard shortcuts, some items may not have a number shortcut (such as the "Others" item in this figure).



Two-handed Fingers Tap

Fig. 2. Finger-Count in expert mode. The user performs a fast two-handed finger tap to execute a command without waiting for the menu display.

We also propose Finger-Count (FC) menus, which allow novice use of FC gestures and act as a teaching method for learning them. This menu technique makes our gesture set visible and easily discoverable. It favors the transition from novice to expert behavior by making active exploration possible with the same gestures being performed in novice and expert modes. It avoids interrupting novice users' tasks and instead supports the user in "learning by doing".

Finger-Count can be investigated from two perspectives: it can either be seen as a multi-touch gesture set or as a multi-touch menu technique. We explore these two perspectives and highlight the properties and limitations of FC interaction. We also present extensions for increasing the number of commands and for allowing multi-user interaction. As an application, we show how easily the Finger-Count technique can be integrated into a 3D modeling prototype, an image manipulation prototype and a professional geographical information system (GIS).

Following a study that shows that Finger-Count gestures are easy to learn compared to a variant of Marking menus, we performed a user study investigating the respective efficiency of the 25 possible FC gestures regarding learning and finger chord preference. Results show that 1) gestures combining one and five fingers as well symmetrical gestures are easy to learn; 2) users adopt two major strategies for performing FC gestures ("Symmetric" and "Left-to-Right").

Finally, we adapted the Finger-Count (FC) concept to *in-the-air* gestures. Using the same principle, the FC technique can hence either be used when touching a surface or when performing gestures remotely. We believe this property is useful for scenarios involving tabletops, interactive television and public displays. While touch interaction appears to be more efficient and comfortable, the ability to support distant interaction can be useful when users just want to execute a quick command without approaching the display.

Our contributions involve:

- The introduction of a coherent and efficient multi-touch gesture set as well as a teaching method for learning them through a menu technique.
- The results of a user study showing that the FC technique is efficient and easy to learn, and exploring the differences between the different possible FC gestures.
- The extension of Finger-Count to in-the-air gestures to favor a fluid transition from touch to touch-less interaction.
- The design, implementation and integration of the FC technique into three different applications.

2. CHALLENGES FOR MULTI-TOUCH GESTURES AND MENU TECHNIQUES

In this section, we provide challenges from both the multi-touch gesture and the menu technique perspectives.

2.1 Multi-touch gestures

Multi-touch gestures have potential for increasing input bandwidth because they allow users to manipulate multiple degrees of freedom (DOFs). However, designing a set of coherent multi-touch gestures is challenging for several reasons [Norman 2010, Wigdor et al. 2011]:

- *Ambiguity*: A novel set of gestures should be compatible with existing and traditional operations (such as the pan gesture or the now common two-finger pinch gesture for zooming).
- *Visibility*: Visual clues should be provided for discovering gestures as “most gestures are neither natural nor easy to learn or remember” [Norman 2010].
- *Coherence*: The gestures of the set should “make sense together” [Wigdor et al. 2011] to easily understand which gestures can or cannot be interpreted by the system.
- *Learning*: Learning a large set of gestures being a difficult task, active exploration of the gesture set should be favored to enhance transition from novice to expert usage.

In addition, various others aspects such as comfort or accuracy must also be taken into account when designing the gesture set.

2.2 Menu techniques

Menu techniques are widespread in current applications for presenting, organizing and selecting commands. Contrary to pure gestural systems, menu techniques exploit recognition rather than recall to reduce mental effort [Lee 1993]: they make possible actions visible and easily discoverable [Norman 2010]. Linear menus (menu bar, pull-down menus, etc.) are a traditional way to select commands on personal computers. However, they suffer a number of drawbacks on touch-sensitive surfaces:

- *Occlusion*. The hand and the fingers may hide parts of the menu.
- *Accuracy*. The large size of the finger contact area may induce item selection errors. Occlusion and accuracy are related to the *Fat Finger problem* [Siek et al. 2005].
- *Lack of shortcuts*. In the absence of a keyboard, the expert mode of linear menus, which is based on keyboard shortcuts, becomes unavailable.

For large interactive surfaces such as tabletops or wall displays, further drawbacks are:

- *Reachability*. Considering the length of the human arm, menu bars can be difficult to reach and traditional menu systems difficult to operate.
- *Multi-user support*. Collaborative work often requires users' identity to be known. Unless using specific technology, this information is missing [Dietz et al. 2001].

The Finger-Count technique has the twofold objective to alleviate the limitations of multi-touch gesture sets and of those related to menu techniques on interactive surfaces. The proposed approach consists in combining multi-touch gestures and menu techniques in order to make gestures self-revealing. Moreover, multi-touch capabilities serve as a means to facilitate access to favorite menu items.

3. RELATED WORK

3.1 Gestural menu techniques

Some interaction techniques designed for the PC combine menu techniques and gestural interaction. Most of them are based on circular menus [Kurtenbach et al. 1991] rather than Linear menus [Appert et al. 2009, Roudaut et al. 2009]. For instance, Marking menus [Kurtenbach et al. 1991] are a combination of circular menus in novice mode and gestural interaction in expert mode. They are very efficient because they favor the transition from novice to expert usage as users perform the same gesture in both modes [Kurtenbach et al. 1991, Zhao et al. 2004]. While Marking menus are a successful self-revealing single-touch gesture system, they do not provide a direct solution for multi-touch gestures [Wigdor et al. 2011]. Marking menus can contain 8-12 items on a single hierarchy level. Several enhanced Marking menus have been proposed to increase the number of menu levels (menu depth) [Kurtenbach et al. 1993, Zhao et al. 2004, Bailly et al. 2007], and/or the number of commands at each menu level (menu breadth) [Zhao et al. 2006, Bailly et al. 2008].

3.2 Multi-touch Interaction

A few menu techniques have been designed for touch interaction such as [Leithinger et al. 2007, Hesselmann et al. 2009, Chaboissier et al. 2010]. We review here the techniques using several fingers or two hands.

Multi-finger menus. Several menus using multi-touch capabilities have been proposed for interacting with tabletops [Koike et al. 2002, Shahzad et al. 2007, Wu et al. 2003]. But most of these techniques permanently display the menu and do not allow pure gestural interaction. In contrast, some recent techniques [Bailly et al. 2008, Chung Au et al. 2010, Lepinski et al. 2010] use all the fingers of the hand to select a command and allow pure gestural interaction for expert users. MTM [Bailly et al. 2008] and the Palm menu [Chung Au et al. 2010] are two multi-touch pop-up menus displaying buttons under each finger for eyes-free interaction. While MTM uses the heel of the hand to activate the menu system, Palm menu requires the user to perform a five-finger tap. These two techniques do not require specific chording gestures. This differs from the Multi-touch Marking menu [Lepinski et al. 2010], a 2-level multi-touch marking menu that requires a chording posture, which is indicated by a permanent ”chord map“. Users must perform a multi-finger stroke in a specific direction to select the desired item. By considering 8 comfortable chording postures and 8 directions, Multi-touch Marking menus can contain up to 8*8=64 commands. As for Multi-touch Marking menus, Finger-Count menus combine menu techniques and multi-touch gestures. However, Finger-Count interaction requires less screen real estate, benefits from two-handed interaction and does not require a technology capable of detecting finger identity to recognize chording gestures.

Two-handed interaction. Multi-touch technologies favor two-handed interaction, which is common in the physical world, increases the parallelism of manipulations and reduces task-switching time [Jiao et al. 2010]. But two-handed interaction may also increase cognitive load [Kabbash et al. 1994]. The Kinematic Chain model [Guiard 1987] provides the three following principles to design efficient two-handed interaction techniques:

- The non-dominant hand (NDH) sets the frame of reference for the action of the dominant hand (DH),
- The NDH takes precedence over the dominant hand,
- The granularity of action of the NDH is coarser than for the DH.

Finger-Count menus follow these three principles.

Finally, most two-handed menu techniques only use the NDH for command selection [Koike et al. 2002, Odell et al. 2004]: the DH manipulates objects of interest or controls some parameters. In contrary, the Toolglass [Bier et al. 1993] really exploits both hands to select commands: the user uses his/her non-dominant hand to control the spatial position of a translucent tool palette and uses one finger of his/her dominant hand to select commands. These techniques do not allow a pure gestural interaction and provide only a limited number of commands. Recently, the Two-Handed Marking menu [Kin et al. 2011] was proposed as an extension of Marking menus for two-handed interaction. But a drawback of this technique is that it consumes drag events and is not compatible with traditional interaction techniques such as pan or "zoom and rotate" interactions.

3.3 Multi-touch gesture teaching

We first present cheat sheets, the traditional method for teaching gestures. We then describe advanced mechanisms that guide users interactively by combining feedforward and feedback.

Cheat sheets. Cheat sheets provide a complete overview of the available commands and their associated gestures. Gestures are generally represented by diagrams / pictograms [Brandl et al. 2008, Finger-Works, Elias et al. 2007], animations [Kurtenbach et al. 1994] or short videos (Apple Mac OS X). Gesture play [Bragdon et al. 2010] proposes using small games (with physics simulation) to facilitate the learning of gestures.

Combining feedforward and feedback. Octopocus [Bau et al. 2008] is an interaction technique combining feedforward and feedback for pen-based interfaces. Feedforward consists in showing the next portion of available gestures. Feedback indicates how the recognition system is interpreting the input. Shadowguides [Freeman et al. 2009] and

Arpege [Bau et al. 2010] extend Octopocus to multi-touch gestures. While Shadowguide mainly focuses on whole-hand gestures and requires a “registration pose guide” for guiding gestures, Arpege mainly focuses on static chording gestures (not requiring tracking) and guides the user step by step.

Indeed these approaches help the user to learn gestures, but they tend to require a large amount of screen real estate for displaying cheat sheets or feedforward and feedback information.

4. FINGER-COUNT MENU

Humans naturally count with fingers to extend cognition (children use fingers as an external memory) or to communicate with others. For instance, finger counting is used by basketball referees to signal the number of the player called for foul to the administration.

Touch Finger-Count. The Finger-Count (FC) technique uses this principle for selecting commands on multi-touch surfaces. Users just need to put a given number *N* of fingertips of each hand in contact with the surface. With two hands, a user can specify 5x5=25 different items. A FC menu bar associates each non-dominant hand (NDH) *N*-finger touch with a pull down menu in the menu bar. The *N* value is displayed next to the name of the item that opens the menu (Figure 1). Likewise, the dominant hand (DH) selects an item in the currently selected menu. So, the user selects an item just by making the appropriate number of finger contacts with each hand. The corresponding command is activated when the user lifts all his fingers. As fingers cannot be lifted up at exactly the same time, “simultaneousness” is defined with some time tolerance (100ms in the current design). In the case of shared or public devices, which cannot be configured by the user, the left hand always acts as the NDH (and opens the menu) while the right hand acts as the DH (and selects the item).

Touchless Finger-Count. We also propose a variant called Touchless Finger-Count (Figure 3) for interacting with a distant display. This technique is based on a similar conceptual model. But instead of touching the surface with fingers, with Touchless FC users must exhibit the appropriate number of fingers with the hands directed towards the surface (Figure 3). A depth camera such as the Microsoft Kinect is then used to recognize the number of fingers by extracting hand contours [Bailly et al. 2011]. While finger-counting has already been used in the context of immersive virtual environments for entering numerical input [Lapouras 2009], Touchless Finger-Count investigates how this idea can be efficiently used for command selection.

Fig. 3. Touchless Finger-Count in the context of interactive television. Left: design. Right: implementation.

We now describe the main properties of Touch and Touchless FC menus. We will discuss the differences between these two variants in section 9.

4.1 Gesture set

Coherence. Finger-Count gestures make sense together because our gesture set forms a "whole" where elements can be ordered by an *ordinal variable*. Our gestures are all of the same kind and have a natural order, which is a rather unusual and interesting property. This makes it possible for users to guess the cover of the gesture set (i.e., which gestures can be understood by the system). Users can also easily guess the previous (by removing a finger) or next gesture (by adding a finger) and thus discover the different gestures and their associated commands step by step. Another important characteristic is that only the final finger configuration is taken into account, which makes our gesture set *commutative* (it does not matter in which order the fingers are put down) [Wigdor et al. 2011]. The only restriction to this rule is that all fingers must be lifted together to execute the command. Finger-Count gestures hence differ from complex chording gesture sets or whole-hand gesture sets which generally provide no straightforward rank ordering and are therefore more difficult to organize in a coherent way in a menu.

Comfort. Finger-Count gestures are easy to perform because they do not require complex hand postures or potentially uncomfortable finger chordings [Bau et al. 2010]. Moreover, a key point is that Finger-Count relies on static gestures: the user does not have to move fingers on the surface, thus avoiding undesirable friction effects and mechanical strain on the skin after prolonged use.

Robustness. Most multi-touch devices can capture 10 simultaneous finger contacts, which is sufficient for Finger-Count gestures (as there is no need for hand-shape detection nor specific chording recognition). FC gestures are also easy to recognize, as the system only needs to count the number of finger contacts for each hand. For single user applications, the surface can be split into two areas (one for each hand). For multi-user applications, local areas can be used as explained in section 6.2. Technologies supporting a hand model such as [Han 2005] make it possible to avoid the use of input areas, as the display knows which contacts belong to which hand. Another reason for robustness is that FC is based on a static hand posture rather than on a dynamic movement. Thus, a single frame is sufficient to recognize the gesture.

Cancel. A common problem with gesture sets is how to cancel a gesture once started. Using FC, users can cancel by removing the NDH from the surface (or by closing it with touchless FC). From a user point of view this is coherent with the fact that no menu is selected (and thus no action performed) if the NDH is not used.

Visibility. One drawback of gestures is that "pure gestural systems make it difficult to discover the set of possibilities" as most gestures are not "natural" [Norman 2010]. Finger-Count provides a way to reveal and logically present a set of multi-touch gestures. The menu technique makes gestures and their corresponding actions visible and therefore easily discoverable.

4.2 Menu technique

Finger-Count menus exploit FC gestures to extend the traditional menu bar. Users can still access the menu bar by pointing at items in the usual way, i.e., by touching them with one finger. However, they can also perform FC gestures to access up to 25 favorite items in the menu. Importantly, this does not impose limitations on the maximum number of items in the menu system: as for keyboard shortcuts, items that do not have a FC shortcut can be activated by pointing at them.

Navigation. Users can navigate in menus just by adding or removing fingers with the NDH. When the user finds the right menu, she can then select the desired item by pressing the appropriate number of fingers with the DH.

Expert mode. FC gestures provide a substitute for the keyboard shortcuts of Linear menus (which are generally not available on interactive surfaces because of the lack of a physical keyboard). As in the Marking menus [Kurtenbach et al. 1991], Finger-Count menus provide an expert mode based on gestural interaction: commands can be selected without opening the menu if they are performed within 300ms. This feature avoids occlusion by the menu. It helps maintaining the user attention on objects of interest rather than on UI components. Users can perform multi-finger taps simultaneously with both hands in expert mode: the system interprets the produced trace as soon as all fingers have been removed. Command activation is then performed in one chunk using a two-handed finger posture [Buxton 1986]. A similar mechanism applies for touchless Finger-Count in expert mode: users just need to quickly show the appropriate number of fingers on both hands, then to close their hands to execute the command.

Screen space. The FC technique only requires a single digit to indicate which gesture corresponds to a given command. Hence, it does not require more screen real estate than keyboard shortcuts. This is a main difference compared to Multi-touch menus [Lepinski et al. 2010] or other methods for teaching multi-touch gestures [Freeman et al. 2009], which tend to consume a large amount of screen real estate for displaying hand posture drawings. In fact, using FC, even the menu bar could be hidden in expert mode to save more space.

4.3 Learning

Active exploration and fluid transition. As said above, users not familiar with FC can interact in the usual way, by touching (or pointing at, for touchless FC) menu items to activate them. Thanks to Number Shortcuts, novice users are encouraged to discover and learn the command-gesture associations when navigating in the menu system. As in the Marking menus [Kurtenbach 1991], an important property regarding the learning of FC gestures is that Finger-Count menus favor the *fluid transition* from novice to expert usage because users can execute the same gesture when the menu is displayed or not. Users can hence learn gestures implicitly using 'muscle memory', just by repeatedly using the menu.

Command-gesture association. Commands with a semantic relationship (such as "Save" and "Save As") are usually located in the same menu area. This spatial organization can facilitate learning. Finger-Count also allows similarities between gestures corresponding to semantically related commands: commands in the same menu share the same NDH configuration and thus only differ by their DH configuration. The correlation between the semantic distance of commands, the spatial distance of the corresponding items and the configuration of FC gestures should thus ease the learning of command-gesture associations.

4.4 Compatibility and direct manipulation

Ambiguity. Finger-Count can be made compatible with common (or even standardized [Norman 2010]) operations like panning/zooming or rotating with one or two fingers on interactive surfaces. First, single-finger contacts are ignored if fingers are moved or removed within 300ms, the amount of time generally used for panning or pushing a button. Second, the Zoom/Rotate command can be integrated in the menu by making it correspond to the first item of the first menu, as shown in Figure 4: The standard two-finger Zoom/Rotate posture hence becomes a specific case of a FC command.

Direct manipulation. Gesture sets are generally used for selecting discrete commands. Finger-Count interaction can also be used for direct manipulation and for controlling command parameters. For instance, after putting two fingers on the surface for selecting the zoom-rotate command as explained above, the user just needs to move fingers to control the amount of zooming or rotation straight away. Selecting a command and controlling its parameters in the same gesture has been shown efficient [Pook et al. 2000, Guimbretière et al. 2005] and this idea can be applied to a large variety of multi-finger operations as illustrated with the prototypes described below.

5. APPLICATIONS

The Touch Finger-Count technique has been implemented on an Immersion multi-touch table based on diffused-illumination technology, with a 72x96cm display. Touchless FC has been implemented on different kinds of large screens and TV screens, using the Microsoft Kinect camera. These devices communicate with the Finger-Count program using the widespread TUIO protocol [Kaltenbrunner et al. 2005] through a network connection. Our program is written in C++ using the Qt toolkit, which provides support for multi-touch input. We developed a library for translating TUIO messages to Qt native multi-touch events. Because our interactive table does not provide a hand model, the screen is split into two areas: one for the NDH and one for the DH. This simple solution is adequate for single user applications.

5.1 3D modeling prototype.

This application, based on OpenFlipper[1], illustrates how Finger-Count interaction can be used to edit and manipulate 3D objects (Figure 4 - left). In particular, the first menu, titled "Geometry" (Figure 4 - middle) contains various manipulation tools such as the "Zoom/Rotate" command, which works as described in section 4.4, and the "Pitch" command that is triggered by putting two fingers of the right hand on the surface (Figure 4 - middle) and which value is controlled by performing a vertical gesture (Figure 4 - right).

---

[1] OpenFlipper (http://openflipper.org) is a flexible geometry modeling library.

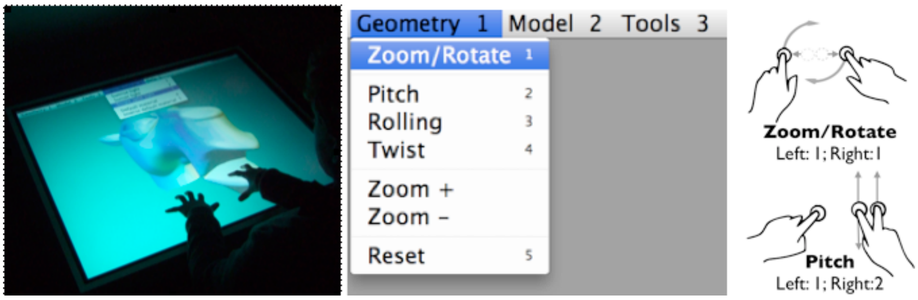


Fig. 4. 3D modeling prototype. Left: One user interacting with Finger-Count. Right: Zoom/Rotate and Pitch gestures. Middle: Their corresponding representation in the menu.

5.2 Image manipulation

This application uses FC for easing image filtering. One key feature is the *Punch filter* that applies a fish-eye effect on a part of the image. Users select this command with four fingers on the left hand and two fingers on the right hand. They control parameters by moving the two right-hand fingers on the surface. These fingers specify both the center and the radius of the fish-eye effect as shown in Figure 5. This prototype illustrates how Finger-Count makes it possible to select a command and control several parameters (x, y location, and radius of the Fish-eye effect) in the same gesture.

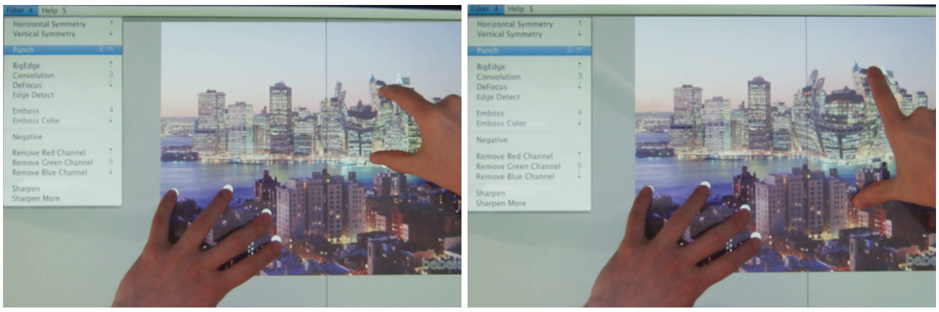


Fig. 5. The user selects the *Punch filter* with the FC shortcut 4-2. The two right-hand fingers specify both the center and the radius of the fish-eye effect.

5.3 Geographical information system (GIS)

Quantum GIS (QGIS)[2] is a professional open source geographical information system, which supports a wide range of features including standard navigation features and advanced tools for editing and analyzing data. Ten face-to-face interviews with GIS users confirmed their interest for adapting QGIS for tabletops (QGIS has been initially designed for desktop computers). From these interviews and our adaptation of QGIS on tabletops, we observed that 25 shortcuts were not sufficient to select all frequent GIS commands and that the Finger-Count technique needed to be adapted for supporting collaborative scenarios and for allowing contextual selections. The next section presents three extensions that alleviate these limitations that have been implemented in the QGIS system.

---

2 http://www.qgis.org/

6. FINGER-COUNT EXTENSIONS

6.1 Supporting more commands

Generally, twenty-five shortcuts are sufficient for most users, as they generally use the few same commands most of the time [Witten et al. 1984]. *Relative Finger-Count* (RFC) gestures [Viard et al. 2011] allow for more commands when needed, typically for expert users. By combining FC gestures with vertical drag gestures they provide up to three time more commands (Figure 6). A RFC menu item must be located above or below an item having a FC gesture. It is activated by first performing the corresponding FC gesture and then moving the fingers up or down. For instance in Figure 6, “Save Project As” is triggered by performing a 2-number gesture (as for “Save Project”) then a vertical up move with the DH. As a reminder for the user, a top or bottom arrow is displayed at the right side of RFC items.

The “Save Project As” and “Save Project” commands semantically derive from each other. Similarly, their related gestural shortcuts also derive from each other. This feature should help users to learn command-gesture associations. Moreover, RFC gestures are compatible with direct manipulation (as described in section 4.4) with the restriction that only horizontal gestures can then be used for controlling command parameters (vertical gestures being consumed by RFC gestures).

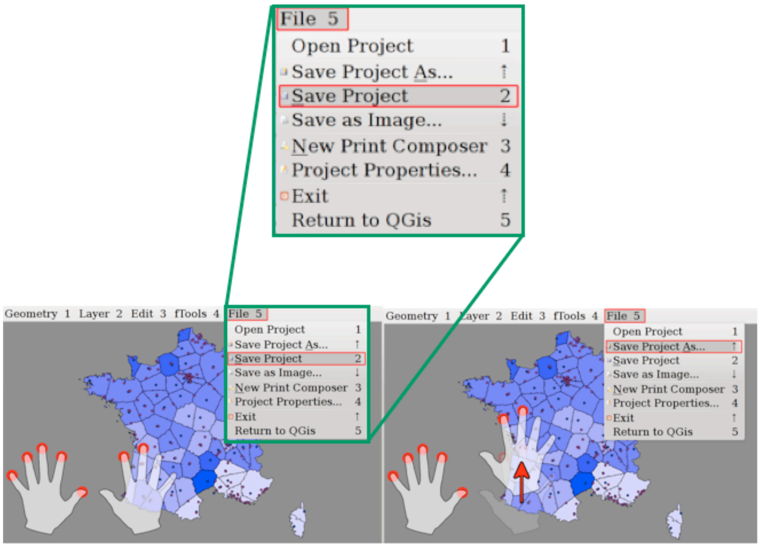


Fig. 6. Using Relative Finger-Count (RFC) gestures to select “Save projects as“ in QGIS.

6.2 Supporting Multiple Users

Finger-Count can be extended to multi-user interaction if the surface provides a hand model (so that the system can identify which pair of hands belongs together). An alternative consists in splitting the screen into dedicated areas (two areas for each user, one for each hand). However, traditional menu bars only support the selection of a single item. *Collaborative Finger-Count* solves this problem by providing a local FC menu to each user, as shown in Figure 7. Each local FC menu provides two pop-up areas. This design enables each user to interact with his own menu system. The size of the pop-up areas is large enough to accommodate up to five fingers.

Fig. 7. Collaborative Finger-Count.

6.3 Supporting Context Menus

Finally, *Contextual Finger-Count* (Figure 8) makes it possible to apply Finger-Count gestures to a specific element of the interface. Collaborative and context FC menus look the same. They differ in the way they are activated: context FC menus appear when users double tap on an interactive element (e.g. a building on a map in Figure 8). The user can then apply commands that are relative to this element by performing FC gestures, thus increasing the total possible number of commands.

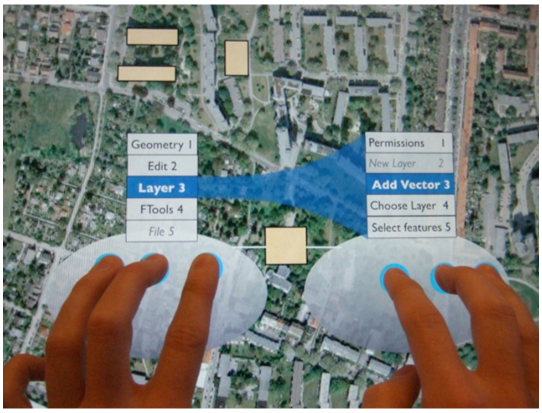


Fig. 8. A contextual Finger-Count menu is activated by double-clicking on the object of interest in the center of the image.

7. PREVIOUS USER STUDIES

In this section we briefly report the main results of two previous studies performed in [Bailly et al. 2010] and [Bailly et al. 2011].

7.1 Touch Finger-Count

The goal of this study was to compare the learning performance of the expert mode of the Finger-Count technique, the traditional "point-and-click" Linear menu, and Radial-Stroke shortcuts, an interaction technique combining radial gestures and linear menus (see [Bailly et al. 2010] for more details). Participants were asked to select as many items (one out of six equifrequent commands) as possible 1) in expert mode, 2) quickly and 3) accurately. For each technique, they performed 4x24= 96 selections.

This experiment provided three interesting findings:

- Users learned the expert mode as well with FC as with Radial-Stroke menus, which have already been shown efficient [Kurtenbach et al. 1991, Bailly et al. 2008].
- The FC technique became rapidly faster than the traditional menu bar after a few training blocks (FC: 1.8s; Menu bar: 2.0s).
- The accuracy of FC was pretty high (greater than 91%). As the experiment focused on learning rather than speed and accuracy, the error rate reflected both motor-control and memorization errors. Accuracy should hence continue to increase with more practice.

These positive results motivated us to perform a second study (section 8) to deeper investigate FC interaction and the differences between the 25 input configurations.

7.2 Touch-Less Finger-Count

In the second study, we compared the Finger-Count, Linear Menu, and Marking Menu techniques, but this time as *in-the-air* gestures. Because performing gestures in the air requires more time, we did not investigate learning performance to keep the experiment less than one hour long. Instead, we compared time and accuracy for the three techniques. Results showed that touchless FC was as fast and accurate as Marking and linear menus (see [Bailly et al. 2011] for more details).

8. USER STUDY

The goal of this study was to compare the difference in learning performance between the 25 different Touch FC configurations and to observe which finger chords were preferred by users. Such results are interesting because they provide guidelines for designers for mapping frequent commands efficiently.

8.1 Menu configuration

The menu bar was hidden by default to force users to learn the complete gesture, i.e. for selecting the menu and the item. The menu bar contained five menus with 5 items each. The list of items was chosen to avoid:

- Possible confusion between categories,
- Particular/direct mappings between text labels and gestures (like Zoom/Rotate),
- Complex or uncommon words.

The order of menus and the order of items in each menu were counter-balanced between participants in order to minimize the impact of textual labels on learning results.

8.2 Task and procedure

Our experiment was inspired by [Bau et al. 2008]. It consisted of 16 blocks of 25 selections. Presentation order for commands within a block was randomized, each command appearing one time. While a zipfian distribution [Witten et al. 1984] would be more realistic [Grossman et al. 2007, Appert et al. 2009], a uniform distribution was chosen to make it possible to compare gestures. The blocks 4-8-12 and 16 were *test blocks* while the 12 others blocks were *training blocks*.

Training block. Each trial begins by displaying the name of the command to select. Users can select commands either in *novice* or *expert mode*. We expected users to perform novice mode selection during the first blocks (as they initially did not know the different command-gesture associations), then to perform more and more expert mode selections with practice.

Once a gesture is performed, a visual feedback indicates the name of the command that was recognized and whether it was the correct command or not. We asked participants to "learn each command and perform the associated gesture as quickly and accurately as possible, trying to improve performance each time", as proposed in [Bau et al. 2008].

Test block. After 3 training blocks, participants performed a test block. Test blocks allowed us to assess whether participants learned commands (even if they did not use the expert mode during training blocks). Participants then performed the 25 selections in expert mode (novice mode was disabled). The stimulus still consisted of the name of the command. There was no feedback related to the selected command, neither if the selection was correct or not. We only displayed the overall score at the end of the test. Instructions consisted in performing gestures as fast and accurately as possible.

Throughout the experiment participants could rest between blocks.

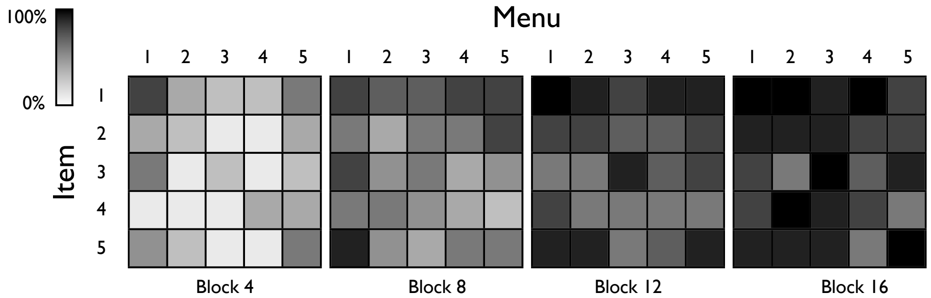
8.3 Participants and apparatus

Fifteen volunteers, 2 females and 13 males, 24 to 36 years old, participated in our experiment. We selected the Apple iPad because this device guarantees a *very* high level of accuracy (no apparent false positives or false negatives). Accuracy was a key point in this experiment: it would have been quite difficult otherwise to distinguish *recall errors* from *recognition errors*. The Apple iPad was used with a separate 15" screen to display the menu and to avoid possible occlusion. In order to let users interact without looking at the input device we added physical borders (plastic tubes) on the iPad so that the users could haptically differentiate the left and right areas.

8.4 Results

Recall. Recall is a binary measure, which is 1 when the participant recalled the right command during test blocks, 0 otherwise. We also imposed that selection time should be less than 5s (typically 2 times longer than for novice mode selection) to prevent scenarios where users would spend too much time to recall commands. The percentage of recall for each item is illustrated in Figure 9.

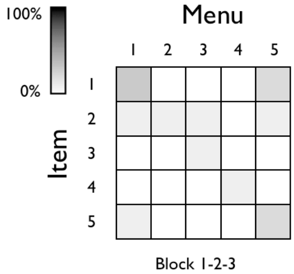
ANOVA revealed a significant main effect for *block* on recall ($F_{3,45}$ = 136, p<.0001). A post-hoc Tukey test (alpha=5%) confirmed that participants learned items with practice (Block B4: 31,1%; B8: 59.5%; B12: 75.3%; B16: 86%). ANOVA also revealed an effect for *FC shortcut* on recall ($F_{24,360}$= 5,19, p<.0001). A post-hoc Tukey test (alpha=5%) showed that the first and last items of the first and last menus, 1-1 (89%); 1-5 (79%); 5-1(79%) and 5-5 (75%), are significantly easier to learn that items 2-3 (45%); 5-4 (46%) and 4-3 (48%).



Item \ Menu	Block 4: 1	2	3	4	5	Block 8: 1	2	3	4	5	Block 12: 1	2	3	4	5	Block 16: 1	2	3	4	5
1	80	40	27	27	60	80	73	67	73	80	100	93	73	87	93	100	100	87	100	87
2	47	20	13	20	40	67	33	60	67	80	80	73	73	67	87	93	93	93	80	87
3	60	7	33	20	27	73	53	60	40	47	67	60	87	67	73	80	67	100	73	93
4	67	13	7	40	40	60	60	53	47	33	80	60	67	60	53	80	100	93	87	60
5	53	27	13	20	53	90	53	47	60	53	87	100	60	67	87	93	87	93	67	100

Fig. 9. Visual (top) and numerical (bottom) representations (top) of the percentage of recall for each command and test block id.

An analysis of the three first training blocks, which have been merged due to the small number of correct selections performed in expert mode, is illustrated in Figure 10. It reveals that except FC shortcuts 1-2 (6%), 3-2 (6%) and 5-2 (9%), only the first and the last item of the first and last menu and the symmetrical FC-shortcuts (those requiring the same number of fingers on each hand) have been performed in expert mode.



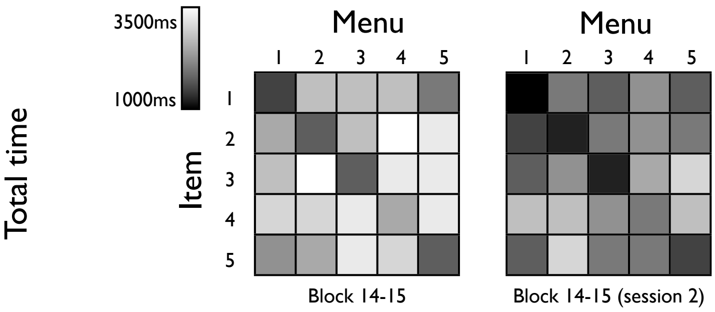
Menu (Block 1-2-3)

Item	1	2	3	4	5
1	30	2	2	2	20
2	9	9	7	4	9
3	4	2	9	0	2
4	0	2	2	7	2
5	16	2	4	0	18

Fig. 10. Visual (left) and numerical (right) representations of the percentage of recall for each command during the first three training blocks.

Total time. Total time (Figure 11 - left) is measured as the time from when the stimulus appears to the time the command is executed (last removed finger). While instructions asked participants to learn each command and to perform gestures as fast and accurate as possible, they confessed focusing more on learning (our main interest) than speed, especially during test blocks. For this reason, we report speed for the two last training blocks (blocks 14-15). ANOVA reveals an effect for *FC-shortcut* on total time ($F_{24,360}$=4.91, p<.0001). A post-hoc Tukey test (alpha=5%) shows that the corner items 1-1 (1.7s), 5-5 (2.1s), 1-5 (2.4s), 5-1 (2.5s) as well as two more symmetrical FC-shortcuts 2-2 (2.5s) and 3-3 (2.2s) are significantly faster than 2-3 (4.2s), 4-3 (4.1s) and 5-4 (4.0s).

We asked three of our participants with both a high level of speed and recall to perform a second session (for a total of 16+16=32 blocks) to have data for more experienced users. The results are illustrated in Figure 11 - right.



Menu (Block 14-15)

Item	1	2	3	4	5
1	1414	2692	2590	2636	2065
2	2394	1722	2870	3540	3207
3	2842	3388	1664	3215	3160
4	3017	2976	3196	2542	3254
5	2164	2496	3298	3036	1882

Menu (Block 14-15 (session 2))

Item	1	2	3	4	5
1	1086	2028	1821	2253	1657
2	1392	1331	1990	2366	2063
3	1689	2131	1346	2577	2880
4	2694	2822	2306	2015	2771
5	1731	2881	2035	2094	1556

Fig. 11. Visual (top) and numerical (bottom) representations of the total time for each command. Left: blocks 14 and 15 (all participants). Right: blocks 14 and 15 of the 3 participants performing a second session.

8.5 Discussion

Recall. Results suggest that FC shortcuts [1-1; 1-5; 5-1; 5-5] (learning rate > 75%) are easy to recall. One possible explanation relates to the "Serial position effect". The "Serial position effect" (also "Primacy" and "Recency" effect) [Ebbinghaus 1913] explains that the first and last items of a list are recalled more frequently than the middle items. In our study, these four FC shortcuts are in the first and the last menu and are also the first and last item of these two menus. Most participants also mentioned that "I first learned categories, then I learned items". Another possible explanation is related to "Singularities" in menu positions. Stimuli that are exceptional within their context attract more attention, and attention supports learning [Anderson 1990]. Our 25 FC shortcuts can be interpreted as a "matrix". The "corners" of the matrix were learned faster. The expression "matrix" was used by 3 participants during the open discussion. Finally, these 4 FC shortcuts could be seen as a combination of "elementary" gestures combining 1 finger or 1 hand (5 fingers).

Symmetrical gestures & learning. The three first training blocks as well the first test block suggest that "symmetrical gestures" are learned first. 13/16 participants mentioned that symmetrical gestures were easier to learn. This is in contrast to FC-shortcuts 1-5 and 5-1, which were never explicitly mentioned as easy to learn: only 4 participants mentioned that the first "row" (4 participants) and the last "row" (2 participants) were easier to learn. Ten participants also mentioned that symmetrical gestures are "easier to perform". The fact that they require less mental effort can favor learning.

Speed. The analysis of the two last training blocks suggests a similar pattern for speed and recall efficiency. As our study focuses on learning, speed is probably related to memory access time and does not reflect performance for "expert usage". However, 12 participants mentioned that symmetrical gestures are faster to perform, possibly because they require less mental effort. However, deeper investigation with a user study focusing on speed and accuracy would be necessary to confirm this assumption and precisely distinguish reaction time (cognitive aspect) from execution time (motor control).

Simplicity. The goal of the experiment did not focus on initial learning, i.e., how users learn the technique without explicit teaching. However, we made interesting observations when explaining the technique, during the instruction phase. Indeed, we noticed that it was sufficient to explain how to select a menu (by using the correct number of left hand fingers) to the participants. They would then understand how the technique works and how to navigate in the menu. These observations are not sufficient to conclude about immediate usability, but they suggest that the technique is simple to understand.

Gesture vs. menu. The originality of Finger-Count is to combine gesture (input) and menu technique (output). Both may impact learning. For instance, the "serial position effect" is more related to the visual organization while the "symmetry" is also a property of the gesture. Further, the learning strategy of the user may impact learning performance. While some participants mentioned to learn a "matrix" of items, it is also possible to associate a command to a pair of numbers (e.g. 3-2), or to directly associate a command to a finger movement if it is highly automated.

With the best our knowledge, a limited number of studies [Bailly et al. 2008; Bailly et al. 2010, Bau et al. 2009] compared learning performance of gesture sets with their teaching technique and none of them compared gesture efficiency within the set. This study is probably one of the first attempting to do this and raises several questions. In particular, questions to investigate are the impact on learning of

- Visual organization
- Individual gesture characteristics
- Global coherence of a gesture set
- Learning strategy, and
- Interference between gestures, or between gestures and their corresponding representation in the teaching method.

The answers to these questions are outside of the scope of this paper and are interesting directions for future work.

8.6 Chording gesture analysis

The hands of each participant were video-recorded during the experiment in order to investigate, which fingers participants naturally used for each FC shortcut.

Neighboring chording gestures. Participants exclusively performed chording gestures involving neighboring fingers. No gesture with at least one lifted finger between two fingers in contact with the surface was executed during the experiment. This confirms that participants naturally prefer neighboring chording gestures.

Chording gesture variability. Participants used different chording for the same FC gesture. For instance, some users express "4" with the index (I); middle (M); ring (R); and little (L) while others used thumb (T); I; M; R as shown Figure 14. We also observed that most participants changed the chording gestures during the experiment. For instance, one user equally used the thumb, index, and middle finger to express "1".

Left vs. Right. Surprisingly, some users (7) did not use the same fingers on the left and right hands while expressing the same digit, as shown in Figure 13. For instance, four participants used the little finger on the left hand and the index finger on the right hand. Two users mentioned "I used the little finger on the left hand because it is on the left side like the first item of the menu bar. However, they found it more natural to use the index of the right hand to express "1". Six of these 7 participants tended to use the same fingers on both hands in the second half of the experiment, especially for symmetrical gestures.

Favorite chording gestures. The two main favorite chording gesture strategies for each FC shortcut during the two last blocks of the experiment are shown in Figures 12 and 13.

The *symmetric* strategy uses the same fingers on the left and right hand and was performed by 6 participants. Participants put fingers in this order: index, middle, ring, little finger and finally thumb.

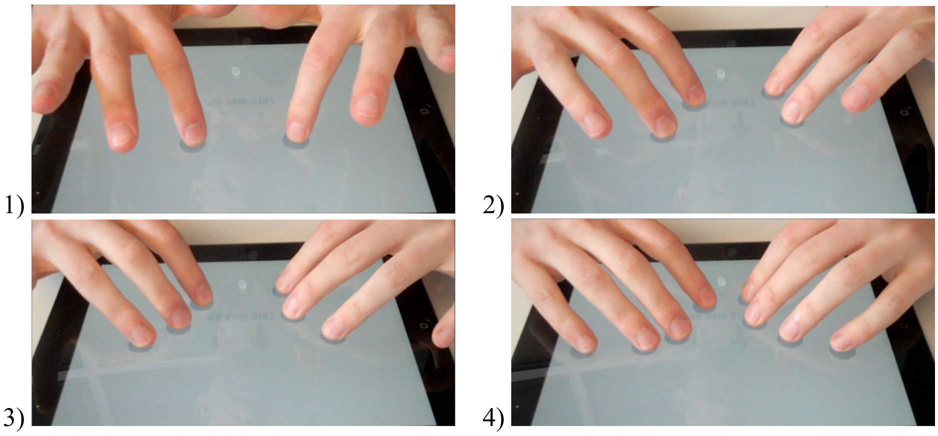


Fig. 12. The “symmetric” strategy illustrated with four symmetrical gestures.

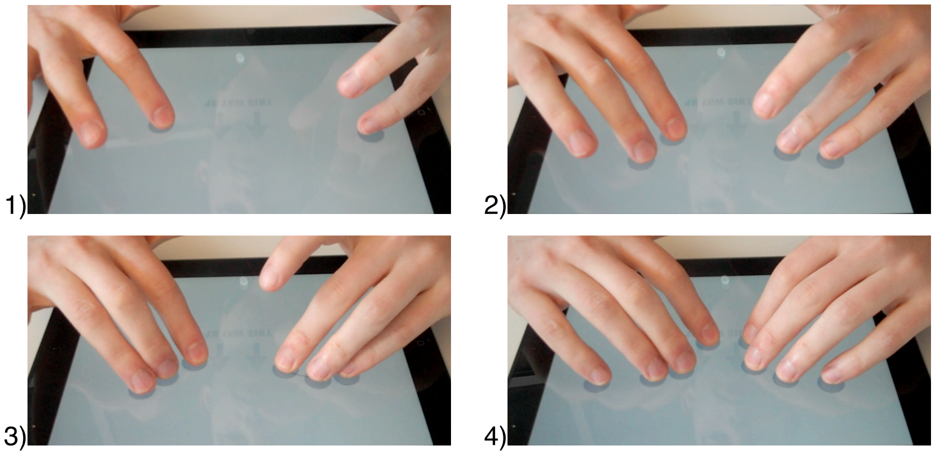


Fig. 13. The “Left-to-right” strategy illustrated with four symmetrical gestures.

The second strategy, (*Left-to-right)*, was performed by 4 users and is illustrated in Figure 13. It is similar to the *symmetric* strategy for the right hand (start from the left finger, except for the thumb). However, on the left hand, users start from the leftmost finger as follows: little, ring, middle, index finger and finally thumb.

Participants mentioned that 3 finger combinations (7 participants) and, surprisingly, 5 finger combinations (5 participants) are more difficult to perform. In contrast, one participant mentioned that 3 fingers is easier, but she was one of the two users using the thumb, index and middle fingers to express “3”: “it is like a tripod” as she said.

Finger awareness. When participants were asked about which fingers they used, their answers were generally different from our observations from post-hoc video analysis. This shows that participants are not really aware of which fingers they use. This can explain why some participants changed chording during the experiment.

	# Fingers (left hand) 1	2	3	4	5	# Fingers (right hand) 1	2	3	4	5
Thumb (T)	5%	10%	10%	5%	100%	5%	10%	15%	5%	100%
Index (I)	60%	50%	50%	100%	100%	80%	85%	95%	100%	100%
Middle (M)	10%	45%	100%	100%	100%	5%	85%	100%	100%	100%
Ring (R)	5%	50%	90%	100%	100%	5%	15%	85%	100%	100%
Little (L)	30%	45%	50%	95%	100%	5%	5%	5%	95%	100%

Fig. 14. Identity of fingers for Touch Finger-Count depending on the number of fingers in contact on the surface and the hand identity. Red lines indicate observed chording gestures.

	# Fingers (left hand) 1	2	3	4	5	# Fingers (right hand) 1	2	3	4	5
Thumb (T)	50%	35%	80%	0%	100%	50%	35%	80%	5%	100%
Index (I)	50%	100%	100%	100%	100%	50%	100%	100%	100%	100%
Middle (M)	0%	65%	100%	100%	100%	0%	65%	100%	100%	100%
Ring (R)	0%	0%	20%	100%	100%	0%	0%	20%	100%	100%
Little (L)	0%	0%	0%	100%	100%	0%	0%	0%	95%	100%

Fig. 15. Identity of fingers for Touchless Finger-Count depending on the number of fingers shown and the hand identity. Red lines indicate observed chording gestures.

9. FROM TOUCH TO TOUCHLESS

9.1 Interaction

To our knowledge, most static gesture sets to date have either been developed for touch interaction or for in-the-air interaction, but do not support both. However, in many situations it may be convenient for users both to interact by touching the surface or remotely by performing in-the-air gestures. This may for instance happen when users walk around in a meeting room during a discussion, using multiple walls and tables. Requiring users to learn and use different paradigms, depending on the interactive situation, would be cumbersome. A single paradigm supporting all situations will certainly make it easier to learn and use interaction techniques.

9.2 Gestures

While Touchless FC is based on a similar conceptual model as Touch FC (counting fingers), it involves different finger movements and is therefore not ergonomically identical. Because the hand posture is not constrained by a surface in the case of touch-less FC gestures, different hand postures are possible. We were thus interested in seeing which hand postures and fingers users would use compared to Touch FC.

Touchless hand posture. In [Bailly et al. 2011] we compared different hand positions. Results showed that participants intuitively show fingers with the hands lifted and the palm facing the display. After experiencing different positions, when sitting participants still preferred to have the palm facing the display, while when standing they preferred to have the back of the hand facing the display.

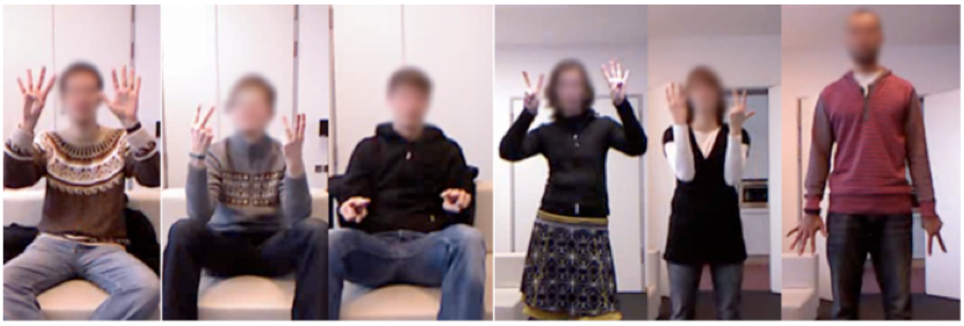


Fig. 16. Various touchless Finger-Count hand postures when sitting (left) and standing (right).

Differences between touch and touchless FC. The fingers used with Touchless FC generally differ from those used with Touch FC (Figure 15). We basically observed two strategies for finger counting (see Figure 17 and 18):

- The "finger-blocking" strategy involved holding down the flexed fingers with the thumb while exhibiting the index, middle, ring and little finger (in this order), with the thumb being used only for the "5" gesture (Figure 17).
- The "non-blocking" strategy involved exhibiting the thumb, index, middle, ring, and little finger (in this order), while flexing the other fingers (Figure 18). Participants were switching between strategies, while most often maintaining the same strategy for the left and right hand.

The "finger-blocking" strategy was used more often for "2", while the "non-blocking" strategy was used more often for "3". We noticed that some participants had difficulties for achieving certain hand postures, e.g. exhibiting three fingers, especially when using the "finger-blocking" strategy. Some of them also reported strain. Most of the participants in this study were Western. While some intercultural differences exist in finger-counting strategies (e.g. many Persians start counting with the little finger [Lindemann 2011]), the Touchless FC technique is compatible with most global finger-counting habits (e.g. Western, Persian, Chinese) for numbers up to 5 per hand.

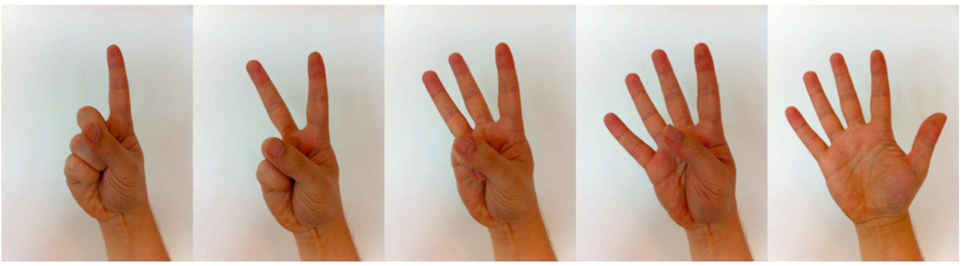


Fig. 17. The "blocking" strategy illustrated with the four symmetrical gestures.

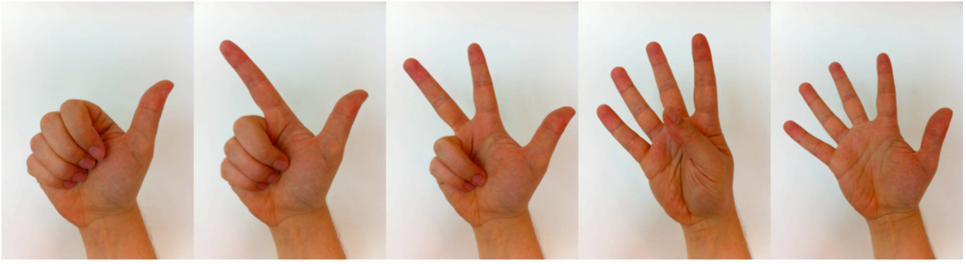


Fig. 18. The “non-blocking” strategy illustrated with the four symmetrical gestures.

From informal observations, users had more difficulty performing Touchless FC gestures (especially showing three fingers) compared to Touch FC gestures and also took longer. We provide possible reasons below.

Possible cognitive reasons. According to our informal observations, participants seem to perform Touch FC in a single step while they often perform Touchless FC in two steps: they extend the correct number of fingers first, then move the hand to the right position. One reason may be that Touch FC provides direct haptic feedback about active fingers (as they touch the surface) while users have to rely on kinesthetic or visual feedback for Touchless FC.

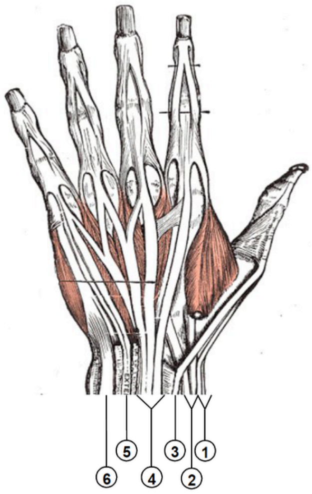


Fig. 19. Left hand from dorsal perspective. The Extensor digitorum communis (4) is used to stretch all fingers at the same time. Fingers that should be flexed must be flexed against its force, introducing strain. (Source: Wikipedia)

Possible physiological reasons. Finger motion is inverted for Touch and Touchless FC: in the first case the indicating fingers are slightly flexed for touching the surface; in the second case they are stretched while the non-indicating fingers are strongly flexed. A closer look at the anatomy of hand reveals important differences in both hand movements. While the thumb uses different muscles, each finger can be flexed individually using two muscles, the deep flexor and superficial flexor. The flexing motion required for touch FC gestures is therefore rather straightforward. Stretching fingers is comparatively much more complex. While the index and little finger can be stretched individually using the Extensor indicis and Extensor digiti minimi muscles, the middle and ring fingers can only be stretched using the Extensor digitorum communis (EDC). This muscle however stretches all four fingers at the same time. If the user wants to exhibit some fingers including the middle or ring finger, while flexing others, he must both activate the EDC (to stretch all fingers) and some other finger flexors at the same time. Because the tendons of the EDC are interconnected (see Fig. 19), this may cause strain on the tendons, depending on how straight the exhibited fingers are intended to be.

In conclusion, while Touch FC and Touchless FC share the same concept (i.e., finger-counting), we observed that, in practice, users did not use the same fingers in both cases. The previous experiment showed that users were not aware of which fingers they used when interacting on a surface, this suggesting that they focused on the number of fingers rather than on the identity of fingers. Finally, we saw that Touchless FC introduces more physiological and cognitive limitations than touch FC. This indicates that touch FC may be very efficient for extensive tasks, while touchless FC may be a useful complement for short tasks, when approaching the surface requires significantly more effort than exhibiting hand postures.

CONCLUSION

In this paper we presented the design and evaluation of Finger-Count (FC) interaction, which combines multi-touch gestures and menu techniques. We showed that the FC gesture set is:

- Simple: participants naturally count with fingers in the real world and they quickly understood how the technique worked during the experiment.
- Robust: gestures can be recognized on most multi-touch surfaces and can be easily performed.
- Expressive: the technique provides a set of 25 gestures, which can be easily extended.
- Coherent: gestures are all of the same kind and they can be ordered.
- Easily discoverable (visibility): Users can navigate in the menu to explore the different command-gesture mappings.
- Not ambiguous with previous techniques. Gestures are compatible with common interaction techniques such as pan, zoom, rotate on interactive surfaces.
- Easy to learn. Gestures are as efficient as radial strokes in Marking menus, which have been proved efficient.

From the perspective of menu techniques, Finger-Count interaction alleviates the problems of occlusion, accuracy, lack of shortcuts and reachability. Moreover, it favors a fluid transition from novice to expert usage. We successfully integrated FC in three pre-existing applications and showed how it can be extended for supporting more commands, multiple users and context menus. However it is worth noticing that the Finger-Counter technique has some limitations of its own that make it inappropriate for some situations: for instance it cannot be operated with only one hand and it is not well suited for very small devices.

Our user study reveals that FC shortcuts combining one and five fingers are especially fast and easy to learn and that symmetrical gestures are promising. This can help designers to map shortcuts to frequent commands. This study also reveals findings such as the two chording gesture strategies (symmetric and left-to-right) or the high variability intra and inter users.

Finally, we presented Touchless Finger-Count and explained that FC can be both used for touch interaction and for in-the-air interaction as they share the same concept. This property allows users to combine the efficiency of touch with the comfort of distant interaction. We highlighted similarities and differences between touch and touchless FC. In particular, users do not use the same chording gestures due to physiological and cognitive constraints. While Touch FC is efficient for intensive tasks, Touchless FC is more useful as a complement for opportunistic scenarios such as public displays, or interactive television

As future work, we plan to investigate questions about the learning of gesture sets that were raised by our user study such as the impact of visual organization, gesture characteristics, and global coherence on learning.

ACKNOWLEGEMENT

This work was partly supported by the Alexander von Humboldt Foundation and the Quaero Programme, funded by OSEO, French State agency for innovation

REFERENCES

- Anderson, J. R. 1990. Cognitive psychology and its implications. San Francisco: Freeman.
- Appert, C. and Zhai, S. 2009. Using strokes as command shortcuts: cognitive benefits and toolkit support. Proc. ACM CHI'09. 2289-2298.
- Bailly, G., Lecolinet, E., Nigay, L. 2007. Wave Menus: Improving the Novice Mode of Hierarchical Marking Menus, Springer INTERACT'07, 475-488.
- Bailly, G., Lecolinet, E. and Nigay, L. 2008. Flower menus: a new type of marking menu with large menu breadth, within groups and efficient expert mode memorization. In Proc. of ACM AVI '08, 15-22.
- Bailly, G., Lecolinet, E. and Guiard, Y. 2010. Finger-Count & radial-stroke shortcuts: 2 techniques for augmenting linear menus on multi-touch surfaces. In Proc. ACM CHI '10, 591-594.
- Bailly, G., Walter, R. Müller, J., Ning T. and Lecolinet E. 2011. Comparing Free Hand Menu Techniques for Distant Displays using Linear, Marking and Finger-Count Menus. Springer LNCS INTERACT'11. 248-262.
- Viard, A., Bailly, G., Lecolinet, E., Fritsch, E. 2011. Augmenting Quantum-GIS For Collaborative And Interactive Tabletops. Advances in Cartography and GISScience, Vol. 1. Lectures Notes in Geoinformation and Cartography, 2011, Part 3, 293-307.
- Bau O., Mackay. W. E. 2008. OctoPocus: a dynamic guide for learning gesture-based command sets. Proc. ACM UIST '08, 37-46.
- Bau, O., Ghomi, E. and Mackay W. 2010. Arpege: Design and Learning of multi-finger chord gestures. Technical report 1533, LRI.
- Bier, E. A., Stone, M. C., Pier, K., Buxton, W., and DeRose, T. D. 1993. Toolglass and magic lenses: the see-through interface. Proc. ACM SIGGRAPH '93, 73-80.
- Bragdon, A., Uguray, A., Wigdor, D., Anagnostopoulos, S., Zeleznik, R. and Feman, R. 2010. Gesture play: motivating online gesture learning with fun, positive reinforcement and physical metaphors. In ACM ITS '10, 39-48.
- Brandl, P., Forlines, C., Wigdor, D., et al. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. In Proc. of ACM AVI'08, 154-161.

- Buxton, W. (1986). Chunking and Phrasing and the Design of Human-Computer Dialogues, Proceedings of the IFIP World Computer Congress, Dublin, Ireland, 475-480.
- Chaboissier, J. and Vernier, F. 2010. CentralMenu: menu partagé sur table interactive. In Proc. ACM IHM'10, 213-216.
- Dietz, P. and Leigh, D. 2001. DiamondTouch: a multi-user touch technology. Proc. ACM UIST'01, ACM, 219-226.
- Ebbinghaus, Hermann (1913). On memory: A contribution to experimental psychology. New York: Teachers College.
- Elias, J., Westerman, W., and Haggerty, M. 2007. Multi-touch gesture dictionary.
- FingerWorks. 2001. igesture products : Quick reference guides
- Freeman, D., Benko, H., Morris, M. R., Wigdor, D. 2009. ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures. In Proc. ACM ITS '09, 165-172.
- Grossman, T., Dragicevic, P., and Balakrishnan, R.. 2007. Strategies for accelerating on-line learning of hotkeys. In Proc.ACM CHI '07, 1591-1600.
- Guiard, Y. 1987. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. Journal of Motor Behavior, 19(4), 486-517.
- Guimbretière, F, Martin, A. and Winograd, T. 2005. Benefits of Merging Command Selection and Direct Manipulation. ACM TOCHI, 12(3), p. 460-476.
- Han, J. Y. 2005. Low-cost multi-touch sensing through frustrated total internal reflection. In Proc. of ACM UIST '05, 115-118.
- Hesselmann, T., Flöring, S., Schmitt, M. 2009. Stacked Half-Pie menus: navigating nested menus on interactive tabletops. In Proc. ITS '09, 173-180.
- Jiao, X, Deng, H., and Wang., F. 2010. An Investigation of Two-Handed Manipulation and Related Techniques in Multi-touch Interaction. In Proc. IEEE MVHI '10, 565-568.
- Kabbash, P., Buxton, W., and Sellen, A. 1994. Two-handed Input in a Compound Task. In Proc. ACM CHI '94, 417-423.
- Kaltenbrunner M, Bovermann T, Bencina R, Costanza E. 2005. TUIO - A Protocol for Table-Top Tangible User Interfaces. In Workshop GW'05.
- Kin, K., Hartmann, B., Agrawala, M. 2011. Two-handed marking menus for multitouch devices. Journal ACM ToCHI'11, 18, 3, Article 16, 23 pages.
- Kin Chung Au, O., Tai, C. 2010. Multitouch finger registration and its applications. In Proc. ACM OZCHI '10, 41-48.
- Koike, H., Xinlei, C., Nakanishi, Y., Oka, K., and Sato, Y. 2002. Two-handed drawing on augmented desk. Proc. ACM CHI'02, 760-76.
- Kurtenbach, G. and Buxton, W. 1991. Issues in combining marking and direct manipulation techniques. Proc. ACM UIST '91, 137-144.
- Kurtenbach G., and Buxton, W. 1993. The limits of expert performance using hierarchic marking menus. In Proc. of INTERACT '93 and ACM CHI '93, 482-487.
- Kurtenbach, G., Moran, T. P., Buxton, W. 1994. Contextual Animation of Gestural Commands. Comput. Graph. Forum 13(5): 305-314.
- Lepouras, G.. 2009. Numerical input techniques for immersive virtual environments. In Proc. IEEE VECIMS'09, 240-245.
- Lee, R., Raymond, D. R., 1993. Menu-Driven Systems. Encyclopedia of Microcomputers, Vol. 11, 101-127.
- Leithinger, D. and Haller, M. 2007. Improving menu interaction for cluttered tabletop setups with user-drawn path menus. IEEE TABLETOP'07. 121-128.

- Lepinski, G. J., Grossman, T., Fitzmaurice G. 2010. The design and evaluation of multitouch marking menus. In Proc. of ACM CHI '10, 2233-2242.
- Norman D. A. 2010 Natural User Interfaces are Not Natural. ACM magazine Interactions 17,3, p. 6-10.
- Odell, D. L., Davis, R. C. Smith, A., Wright, P. K. 2004. Toolglasses, Marking Menus, and Hotkeys: A Comparison of One and Two-Handed Command Selection Techniques. Proc. ACM GI'04, 17-24.
- Pook, S., Lecolinet, E., Vaysseix, G., and Barillot, E. 2000. Control menus: excecution and control in a single interactor. Proc. CHI '00, ACM Press 263-264.
- Roudaut, A., Bailly, G., Lecolinet, E., and Nigay, L. 2009. Leaf Menus: Linear Menus with Stroke Shortcuts for Small Handheld Devices. In Proc. Springer, IFIP TC INTERACT'09, 616-619.
- Shahzad M., S. 2007. An Exploration of Multi-finger Interaction on Multi-touch Surfaces. Ph.D. Thesis. University of Toronto.
- Siek, K. A., Roger, Y. 2005 Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs. Proc. Springer LNCS INTERACT'05, 267-280.
- Wigdor, D., Wixon, D. 2011. Brave NUI World: Designing Natural User Interfaces for Touch and Gesture, Morgan Kaufman, 264 pages.
- Witten I., Cleary, J., and Greenberg, S. 1984. On frequency-based menu-splitting algorithms. International Journal of Man-Machine Studies 21(2). 135-148.
- Wu, M. and Balakrishnan, R. 2003. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. Proc. ACM UIST'03, ACM, 193-202.
- Zhao, S. Balakrishnan, R. 2004. Simple vs. compound mark hierarchical marking menus. Proc ACM UIST'04, 33-42.