



HAL
open science

Studies and guidelines for two concurrent stroke gestures

Alix Goguey, Michael Ortega

► **To cite this version:**

Alix Goguey, Michael Ortega. Studies and guidelines for two concurrent stroke gestures. International Journal of Human-Computer Studies, 2023, 170, pp.102942. 10.1016/j.ijhcs.2022.102942 . hal-04031673

HAL Id: hal-04031673

<https://hal.science/hal-04031673v1>

Submitted on 17 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Studies and Guidelines for Two Concurrent Stroke Gestures

Alix Goguey^a, Michael Ortega^{a,b,*}

^a*Université Grenoble Alpes, Laboratoire d'Informatique de Grenoble*
^b*CNRS*

Abstract

This paper investigates thumb-index interaction on touch input devices, and more precisely the potential of **two concurrent stroke gestures**, *i.e.* gestures in which two fingers of the same hand concurrently draw one stroke each. We present two fundamental studies, one using such gestures for two-dimensional control, by precisely drawing figures, and the other for command activation, by roughly sketching figures. Results give a first analysis of user performance on 35 gestures with a varying complexity based on numbers of turns and symmetries. All 35 gestures, were grouped into six families. From these results we classify these families and propose new guidelines for designing future mobile interfaces. For instance, favouring **anchored** gestures (forefinger drawing while the thumb remains still on the surface) to increase input bandwidth when forefinger precision is required.

Keywords: Touch interaction, User study, Stroke gesture

1. Introduction

Multitouch displays are ubiquitous. In the span of a decade, they have entered our daily lives and became unavoidable. Whether with our phones, in our cars or at a cash register, we use them in an extremely wide variety of tasks and contexts. The trend
5 towards all-touch is even likely to be reinforced, since the use of touchscreens in the professional context is no longer anecdotal¹. As a consequence, we keep adding tasks

*Corresponding author

Email addresses: alix.goguey@univ-grenoble-alpes.fr (Alix Goguey), michael.ortega@imag.fr (Michael Ortega)

¹Wikipedia: Fourth Industrial Revolution

that are becoming more and more complex to perform, and start using tools originally designed for desktop computer on touchscreen (e.g., spreadsheets, text editors, ...). However, the interaction bandwidth on touchscreen is far narrower than on desktop
10 computers. The vast majority of our daily interactions only rely on touches' position. Screen sizes are also becoming smaller. This combination of factors can lead to mobile applications being purposefully limited compare to their desktop counter-parts. For instance, Adobe Photoshop (CS6) offers over 600 different commands on its desktop version and only 35 on its tablet version (Express) [1].

15 It is crucial to think now about tomorrow's multitouch interactions and avoid daily use of limited or non-adapted tools. It therefore comes as no surprise that an extensive amount of research in HCI is dedicated to explore and enlarged the interaction bandwidth of multitouch devices. To avoid the extensive use of the limited amount of screen real estate, or the extensive use of menus, a promising solution to increase the interaction
20 bandwidth is to enrich the input vocabulary. Many works study new dimensions such as the orientation [2, 3, 4], the force exerted during a contact [5, 6], or which finger [7] or hand part [8, 9] is interacting. However, while these new pieces of information look promising and exiting, almost no commercially available device² offer them.

Yet, there are ways to enrich the input vocabulary without any new technological
25 breakthrough. Sets of stroke gestures have been studied to issue commands [10, 11, 12, 13], for instance left and right swipes to display next and previous pictures. More complex gestures, like waves, circles, or spirals have also been designed even though used less often on commercial devices [10]. The number of contacts on a screen has also been used to enrich the input vocabulary, via simple chords [14] or more
30 complex continuous gestures as the well known RST (Rotate-Scale-Translate) gesture [15] typically implemented in picture gallery or map applications. However, while gestures and number of contacts have been combined at times, we argue that there is a gap between simple (*e.g.* two-fingers down swipes to invoke the settings menu on Android) and complex (*e.g.* RST) two concurrent finger gestures that has not been

²Force sensing was available on iPhone (from 6s to XS Max) via Apple 3D Touch technologies but was removed later on.

35 studied. Filling this gap could potentially provide new gesture-based inputs to designers, and improve the expressivity of all off-the-shelf multitouch devices without the need for any additional hardware.

In this work, we focus on this gap and investigate **two concurrent stroke gestures** (2CSG), *i.e.* gestures in which two fingers of the same hand concurrently draw one
40 stroke each. Our goal is to provide an exploration on whether such gestures are usable or not, and therefore if adding a second contact (still or in motion) while already stroking with one finger can act as a modifier. As the forefinger is the most used and studied finger [16] for single-finger gestures, we focus on thumb-forefinger gestures. Furthermore, from the literature we know that, due to mechanical and neuromuscular constraints, the
45 thumb and forefinger are a lot more independent than any other pair of fingers [17]. Such an advantage could be leveraged for a more widespread use of 2CSG.

Our investigation is made of two separate experiments, designed to answer two main questions: 1) can we easily perform 2CSG and 2) can 2CSG be recognized by a system. As the number of possible gestures is virtually infinite, we first defined an atomic set of
50 strokes that could, if combined, represent a wide variety of 2CSG. As shown below, the resulting 2CSG can be grouped in five families: **anchored**, **parallel**, **one-line**, **mirror** and **misc.**, that we compare with a **baseline** family, made of single stroke gestures. The results of our analysis allow us to derive guidelines to exploit the potential benefits of 2CSG in common situations. One takeaway, is the good performance of the **anchored**
55 family (*i.e.*, adding a static thumb while performing strokes with the forefinger) which is a good option to increase input bandwidth when forefinger precision is required.

Our work investigates whether a simple gesture system that can work on any touch devices without requiring special hardware could be achieved. It proposes two main contributions: 1) two experiments that investigate the potential benefits and drawbacks
60 of thumb-forefinger multitouch gestures, and 2) guidelines on the use of such gestures. After presenting the related work, we detail each experiment and its quantitative and qualitative results. These results are then discussed, and used to derive guidelines.

2. Related Work

When it comes to multi-finger gestures, research mainly focuses on the continuous control of an integral task. As defined by Jacob *et al.* in 1994 [18], an integral task is a manipulation of several attributes that combine perceptually. For instance, x and y coordinates of a position, or value and chroma of a color are perceived integrally, while size and lightness are not [19]. In 1985, Krueger *et al.* introduced the early version of what is known today as the RST (Rotate Scale Translate) technique [20], and Moscovich *et al.* were among the first to investigate this concurrent and continuous control with two-fingers [21]. In 2008, they compared uni-manual and bi-manual versions [22]. They proved that uni-manual (thumb + forefinger of the same hand) outperforms bi-manual (one forefinger per hand) in integral tasks. They showed that users were capable of simultaneously adjusting the orientation, spread, and position of the thumb and forefinger of the same hand on touch-screens to control rotation, scale and translation (RST) of a 2D object. Knoedel *et al.* compared indirect and direct versions and showed that direct mapping shortens completion times while indirect mapping provides a better precision. In both cases, users' trajectories are comparable [15]. This RST gesture has had many variations [23, 24] and is one of the most iconic use of two-fingers interaction on tactile surfaces, mostly used to manipulate 2D content like maps or images. However, in RST gestures, only the difference between the starting and the end states matters. The output is simply an interpretation of this difference to update virtual object characteristics (*e.g.* size or position). The path has little importance, which does not correspond to our objective of studying the expressivity of 2CSG.

2.1. Two Concurrent Stroke Gestures: 2CSG

An extensive amount of research focus on single-stroke gestures (*e.g.* [25, 26, 27, 10, 11]). For example, in 1994 Kurtenbach *et al.* proposed a study with marking menus, in which the direction of the stroke determines a menu item [25]. In 2009, Appert *et al.* studied 16 'stroke shortcuts', *i.e.* shortcuts triggered by simple single strokes, like a line, a pig-tail, a 'w' or a two-turns spiral [10]. In 2010, Li *et al.* triggered shortcuts with more complex drawings: a user could draw several strokes, but sequentially and with one finger only [11].

In comparison, very few works study multi-strokes gestures. In 2011, Banovic *et al.* proposed a *Multifinger Pie Menu Interaction* [28]. The technique relies on chords
95 counting the number of fingers in contact with the surface, as in FingerCount [14]. Furthermore, relative positions between the contacts are interpreted, as in Arpège [29], and as such it paves the way towards 2CSG. In *Multifinger Pie Menu Interaction*, each finger has its own role: one anchors the menu while the second selects an item. The interaction has been tested with several finger combinations and timings. Results showed
100 that an anchored thumb combined with a forefinger, middle or ring finger was among the best finger combination, and while chording simultaneously (as opposed to sequentially) was more error prone, it did yield faster time.

With *Pin-and-Cross*, Luo *et al.* made one more step towards 2CSG [30]. This technique combines one or more static touches (*i.e.* “pins”, usually made with the
105 forefinger), with a stroke gesture (*i.e.* “cross”, usually made with the middle finger). The stroke gesture crosses a radial target which issues a command. Both fingers performing a *Pin-and-Cross* are from the same hand. Authors compared a *Pin-and-Cross* contextual menu to a Marking Menu and partial Pie Menu: *Pin-and-cross* is 27% faster. Even though, the stroke part is a simple straight line, this work provides a promising glimpse
110 on the potential use of 2CSG.

In their 2014 experiment, Rekik *et al.* gave the very first elements on the difficulty of drawing figures with two or more fingers [31]. They showed that, although gestures produced with multiple fingers are larger in size and take more time to perform compared to single-touch gestures, *two-finger gestures should be equally exploited, as they were*
115 *perceived not more difficult to produce*. Unfortunately, almost all of the multi-strokes gestures proposed in the experiment, such as the *ball* or the *butterfly tie*, were not achievable with concurrent stroke gestures of single-hand fingers. Only the *double line* figure could have been performed with two concurrent strokes of the same hand, but it has not been tested. Therefore, there are no results on the use of 2CSG. More
120 recently, Leiva *et al.* proposed a model for predicting the expected user completion time of multi-strokes gestures [32]. They conducted an experiment on 82 figures. However, as in the previous study, none was crafted for concurrent stroke gestures.

Finally one could argue that if all fingers stay together and follow the same trajectory,

study results on single-stroke gestures could stand for multi-stroke gestures. For instance,
 125 two-finger swipes used on recent tablet devices to control multitasking, and single-finger
 swipes have very similar time of execution [33]. These types of multi-finger gestures
 manipulate one attribute at a time (the trajectory), and as such, the number of fingers
 acts as a modifier (*e.g.* single-finger bezel swipe down to show android notification and
 two-finger bezel-swipe down to show the parameters).

130 The research literature shows benefits on using two-points gestures, one-point-one-
 stroke gestures, and two-fingers control. However to the best of our knowledge, there is
 little to no results on the potential of 2CSG, or that could directly be transferred from
 studies on single-stroke gestures to 2CSG.

3. Gesture Set and Notation

135 The number of strokes is virtually infinite as many parameters come into play. To
 design gestures representative enough while keeping the number of variations manage-
 able, we defined a set of 35 strokes, basing difficulty on the 1982 Viviani and Terzuolo’s
 conclusion that the time needed to complete a drawing decreases with the radius of
 curvature [34]. We therefore varied the radius of curvature between the strokes and
 140 within the strokes (*i.e.* inflection points). As a result, we used straight lines (—), half
 circles (⤴ and ⤵) and waves (⤶ and ⤷).

We combined these strokes to create 25 combinations in which *UpStrokes* are dedi-
 cated to the forefinger while *DownStrokes* are dedicated to the thumb. For completeness,
 we added two *DownStrokes*: \otimes , acting as a baseline in which the forefinger performs a
 145 single stroke with no contact of the thumb; and \bullet , using the thumb as an anchor (*i.e.* a
 fixed contact) while the forefinger performs a single stroke. In the end, our studies use a
 total of 35 combinations (see Figure 1).

In this paper, we use the following notation to refer to a combination of strokes: $\frac{S_u}{S_d}$
 = $\frac{UpStroke}{DownStroke}$. For instance, \otimes refers to a \curvearrowright drawn by the forefinger and no thumb in
 150 contact (\otimes), \curvearrowright refers to a \sim drawn by the forefinger and a — drawn by the thumb.

All the 35 combinations can be grouped in the following 2CSG families:

baseline: only one stroke drawn by the forefinger.



anchored: one stroke drawn by the forefinger while the thumb remains still on the surface.

155



parallel: the forefinger and the thumb draw the same strokes.



one-line: one of the two strokes is a straight line.

160



mirror: the forefinger and the thumb draw mirrored strokes.



misc.: the forefinger and the thumb draw unrelated strokes, and no straight line.



165 4. Hypothesis

We decided to construct our gesture set from a single-stroke-shape difficulty point of view (from Viviani and Terzuolo's research [34]), and derived all possible combinations which allow us to group them based on their conceptual similarities. We did not use existing taxonomies to derive our families to avoid clustering *difficult* stroke combination (as shown below). However, we use them to help us form hypothesis on user performance.

170

Freeman et al. [35] proposed a taxonomy (built on [36]) which can be used to classify our gesture set prior to our study, into three categories:

175

- **{single-finger, static, path}**: it includes **baseline** gestures, because they use only one finger, and the hand translates while its pose remains the same all along the path;
- **{multi-finger, static, path}**: it includes **parallel** gestures, because they use two fingers, and the hand translates while its pose remains the same all along the path;

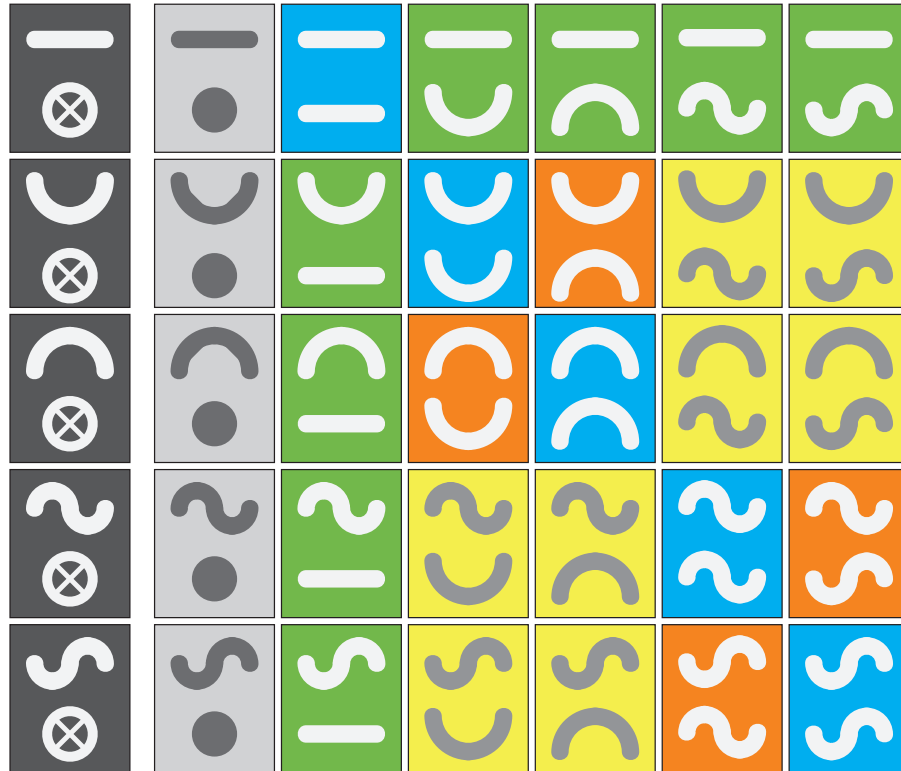


Figure 1: The 35 stroke combinations used in both studies. Top strokes represent forefinger movements. Bottom strokes represent thumb movements. The left column shows the baseline, *i.e.* a single finger stroke made with the forefinger while the thumb does not touch the surface. In the second column, the thumb is anchored effectively acting as a still contact. Background colors each represent a different 2CSG family, and will be used as identifier in the paper.

- *{multi-finger, dynamic, path}*: it includes **anchored**, **one-line**, **mirror** and **misc.** gestures, because the hand translates while the relative position of the fingers changes.

Hypothesis 1 (**H1**): We expect decreasing performance between **baseline**, **parallel** and the remaining families.

Hypothesis 2 (**H2**): **baseline** and **parallel** families only differ from one property (*i.e.*, single- vs multi-finger). Since only one contact is added (*i.e.*, the thumb), we hypothesized that gestures from the **parallel** family should only imply a small mechanical constraint and very low additional cognitive load since both strokes are identical. We therefore expect **baseline** and **parallel** to have similar performance.

More recently, Rekik et al. introduced a taxonomy with the parallelism and sequentiality concepts [37]. However, the induced classification is trial-dependent since the strategy used while performing the gesture is an important feature for the categorization process, and can result in one stroke-combination being potentially categorized differently after each trial. In this taxonomy, two of our gesture families can be classified in unique categories (**baseline** and **anchored**) but none of the other families.

- *{E, 1H, M, 1F, 1F}*: it includes **baseline** gestures, because they are elementary atomic movements (E), made with one hand (1H), a moving forefinger (M) per stroke (1F) and per hand (1F).
- *{C, 1H, R*M, 2F, 1F}*: it includes **anchored** gestures, because they are compound gestures (C) that uses two fingers (2F) of the same hand (1H), one finger per stroke (1F), a static thumb (R) with a moving forefinger (M), which results in parallel gestures (R*M).
- *{C, 1H, M*M or M+M, 2F, 1F}*: it includes the remaining families, because they are compound gestures (C) that uses two fingers (2F) of the same hand (1H), one finger per stroke (1F), a moving thumb (M) and a moving forefinger (M). However, both fingers can move in parallel (M*M) or sequentially (M+M).

Hypothesis 3 (**H3**): Rekik et al. observed several users' strategy from their study. In particular, they relate that "participants combined the movements of their fingers

simultaneously, in [...] symmetric” gestures. They also relate parallelism when fingers
210 make the same atomic movement. We can therefore hypothesize that the two strokes of
both **parallel** and **mirror** families should be mostly drawn simultaneously, while the
strokes of both **one-line** and **misc.** should be drawn sequentially.

We cannot make any assumptions on a correlation between performance and strate-
gies. For instance, using parallelism instead of sequentiality let one think she may save
215 time, but there is no guaranty on movement speed nor the induced cognitive load. Even
if both **mirror** and **parallel** families most likely would use parallelism strategies, we
thus cannot hypothesize on their performance against **one-line** or **misc.** based on the
strategy prism only.

Hypothesis 4 (**H4**): Because of the mechanical constraint of a fixed finger, **anchored**
220 gestures should be slower to perform than the baseline. However, once fixed the anchor
takes a very small cognitive bandwidth. We therefore hypothesize that performance
of **anchored** gestures should be slower but quite close to **baseline**, and faster than the
other families.

225 Hypothesis 5 (**H5**): Given **H2** and given **H4**, we hypothesize that **anchored** and
parallel will yield similar performance.

5. Study 1: Precision

This study aims at evaluating how difficult to draw is a 2CSG. We particularly
observe (1) drawing performance, (2) how stroke complexity and combinations affect
230 parallelism and drawing strategies, and (3) how concurrent movements made by the
thumb affect forefinger drawings.

The experiment consists in a serie of trials in which participants are asked to
reproduce 2CSG. As described previously, each trial prompts a combination of 2
concurrent strokes: five *UpStrokes*: —, ∪, ∩, ∪∩ and ∩∪; and seven *DownStrokes*: —,
235 ∪, ∩, ∪∩, ∩∪, ● and ⊗.

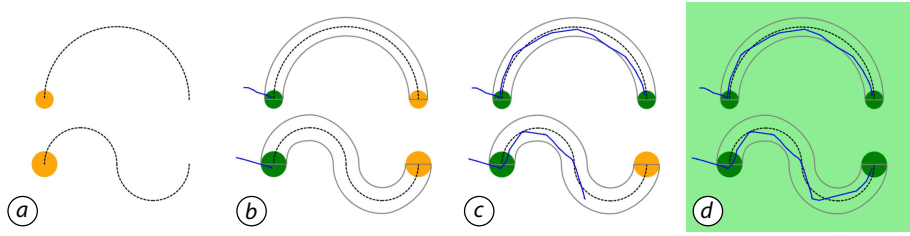


Figure 2: Example of a trial from study 1. *a*: the initial display consists of two starting areas, with different diameters, and the stroke skeletons. *b*: blue strokes display first touch events, *i.e.* finger trajectories. When a participant slid her fingers into the starting areas, the areas turned green, and tunnels and ending areas appeared. *c*: the forefinger reached the ending area of the upper stroke, which turned green, while the thumb is still sliding into the bottom tunnel. *d*: both fingers reached their ending areas. The application background turned green, indicating that the trial was completed.

5.1. Combination sizes

Figure 3 shows the interface used by the participants. The combinations to complete are displayed in the center of the display with dashed-lines. In order to impose a level of precision, we displayed tunnels around the lines.

240 To determine the drawing width, we use the following constraints:

1. ensuring a capped finger separation of 90mm (empirically found to be the maximum comfortable thumb-forefinger distance).
2. no tunnel overlapping (at worst, only the external 1mm thick tunnel strokes would overlap).
- 245 3. using perfect shapes (semi circles).
4. vertically centering strokes on their respective drawing zones.

Following all four constraints, the dashed-lines start and end points are therefore spread 75mm apart on the horizontal axis. We tested 2 tunnel *Widths* (except for ● and ⊗): 10 and 14mm (50 and 70 pixels respectively). The tunnel *Widths* are also used as diameters
 250 of the starting and ending area circles, respectively on the left and right side of the strokes. For ●, participants were required to keep their thumb inside the starting area while the forefinger was completing the *UpStroke*. For ⊗, the thumb was not in contact

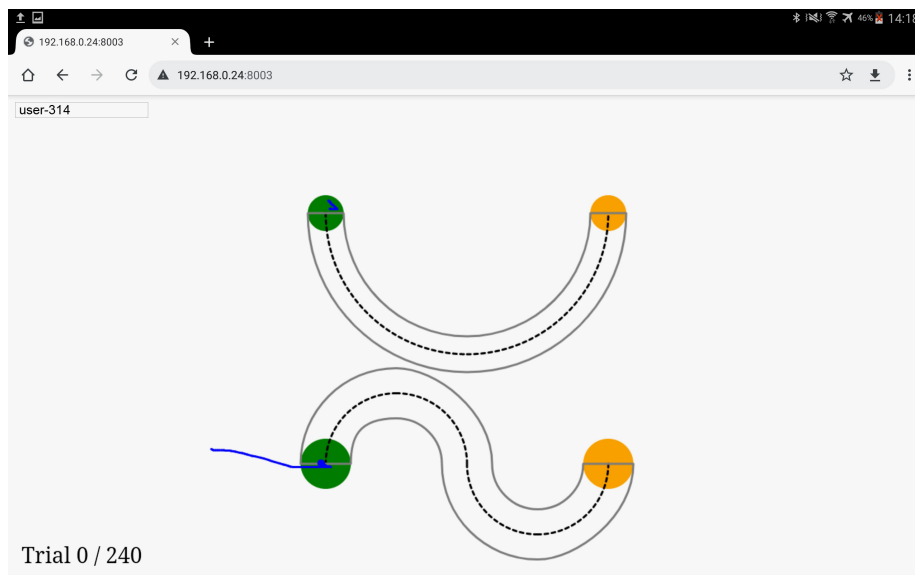


Figure 3: Interface of study 1. Progression is displayed on the bottom-left corner. The two-strokes combination is centred, displayed with dashed-lines, and surrounded by tunnels of different widths. Blue lines represent the strokes that have been drawn by the participant. In the example, both the forefinger and thumb have entered and are in their starting areas (green).

with the surface.

5.2. Procedure

255 After completing a short demographic survey, participants were presented with the touch interface on which they would perform the trials. The device was placed in front of the participants, laid flat on a table. While describing and showing several 2CSG combinations, the experimenter gave the following instructions:

- Participants were asked to draw with their dominant hand, using their forefinger 260 for the *UpStrokes* and their thumb for the *DownStrokes*.
- For each trial, a participant that was not satisfied (*e.g.* bad performance or bad strategy) could lift the fingers and redo the trial as many times as needed. Our rational was to approach as much as possible ideal performance.
- To complete a trial, participants needed to draw both the *UpStroke* and *DownStroke* 265 from their respective starting area to their respective ending area, and keep as much as possible the fingers in their tunnels. Our rational was to avoid frustrating participants when faced with hard combinations.
- A trial started when both the forefinger and the thumb were placed inside their respective starting areas (only the forefinger for \otimes), and ended when both fingers 270 were in their respective ending areas (only the forefinger for \otimes).
- Participants were asked to complete the trials without any constraint on their strategy, as fast and accurately as possible, as long as both fingers were pressed down on the surface. For instance, one could complete the *UpStroke* first and then the *DownStroke*, or on the contrary one could draw both *Strokes* simultaneously 275 (see Figure 2).

After the instructions, and before starting the recordings, participants went through a random serie of trials to get familiar with the task.

All the above phases typically lasted less than 5 minutes. Then participants went through all the trials, and could take a break if needed after each trial. After the 280 completion of all the trials, the experimenter conducted a debriefing session, in which he

asked about the strategies participants developed, and the difficulties they encountered. In total, the experiment lasted from 30 to 45 minutes.

5.3. Experimental Design

Each combination of strokes and tunnel widths was repeated twice. The experiment
285 used a $5 \text{ Ups} \times 2 \text{ Widths} \times (5 \text{ Downs} \times 2 \text{ Widths} + 2 \text{ Downs}) \times 2 \text{ Repetitions}$ within-
subject design. The order of the **240** trials was randomized with the constraint that 2
Repetitions of the same factor combination could not appear twice in a row. A total of
3360 trials were logged across all participants.

5.4. Participants and Apparatus

290 We recruited 14 participants (mean age 27.71 years, SD=3.91 years; no left-handed;
4 females).

All the participants owned a smartphone and used it everyday. 4 participants owned
and used a tablet everyday. Participants barely used two-finger interactions in their daily
life, apart from RST gestures in maps and picture applications.

295 The study was conducted on a 1st generation 11 inches iPad pro (Retina display: 2388
 \times 1668 px resolution at 264ppi). The software was written in HTML/CSS/JavaScript.
Logs were sent through a web-socket to a python-written server running on a MacBook
Pro 13inch fifth generation (Intel Core i5).

5.5. Results and first analyses

300 This section presents both quantitative and qualitative results. For quantitative
results, we explore (1) basic performance, with completion times and success rates, and
(2) concurrency *i.e.* how much strokes are made concurrently. A successful trial means
that the two concurrent strokes were fully drawn within the tunnel boundaries during the
ideal attempt. For qualitative results, we explore (1) fatigue and discomfort, (2) drawing
305 strategies, and (3) perceived complexity.

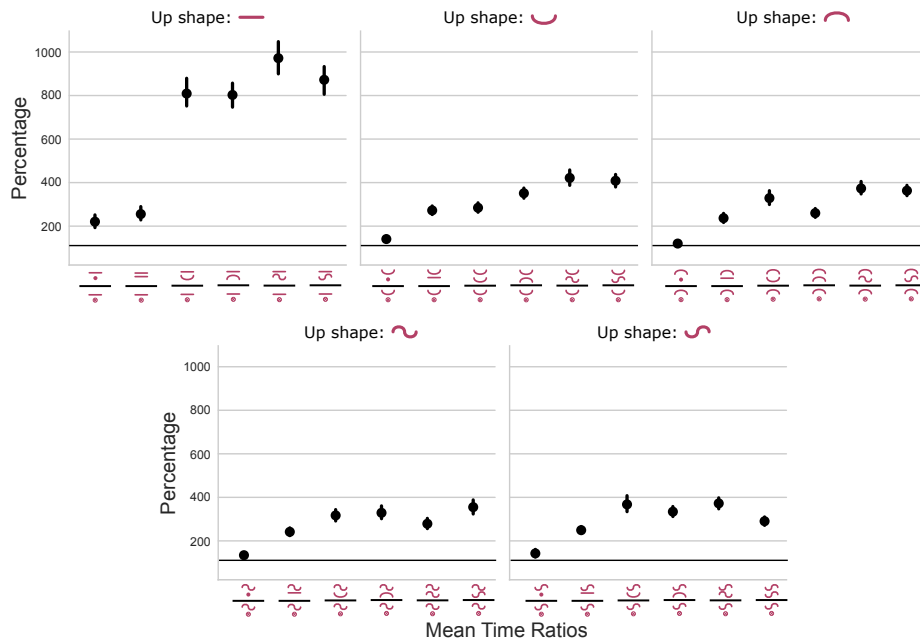


Figure 4: Mean time ratios and 95% CI for all the different combinations of *Strokes*. Ratios correspond to the mean of: completion times for a given combination divided by the mean completion time of the corresponding baseline.

5.5.1. Performance

We computed the mean trial completion time for the 35 combinations, and we also broke it down for each tunnel *Widths* combinations. We performed the same calculations for success rate.

310 Following Dragicevic’s advice on statistical communication for HCI [38], we use estimation methods to derive 95% confidence intervals (CIs) rather than traditional null hypothesis statistical testing. Figures 4 and 5 shows the mean ratios as percentages and 95% CI of time and success rate for each combination of *Strokes* with their respective
 ⊗ conditions (*i.e.* **baseline** conditions).

315 As expected, we observe in Figure 6 that the tighter the tunnels the higher the completion times and the lower the success rates.

As expected as well, we observe an increase in time and error as more complex *Downs* are introduced (Figure 4). When a second shape is introduced, completion time

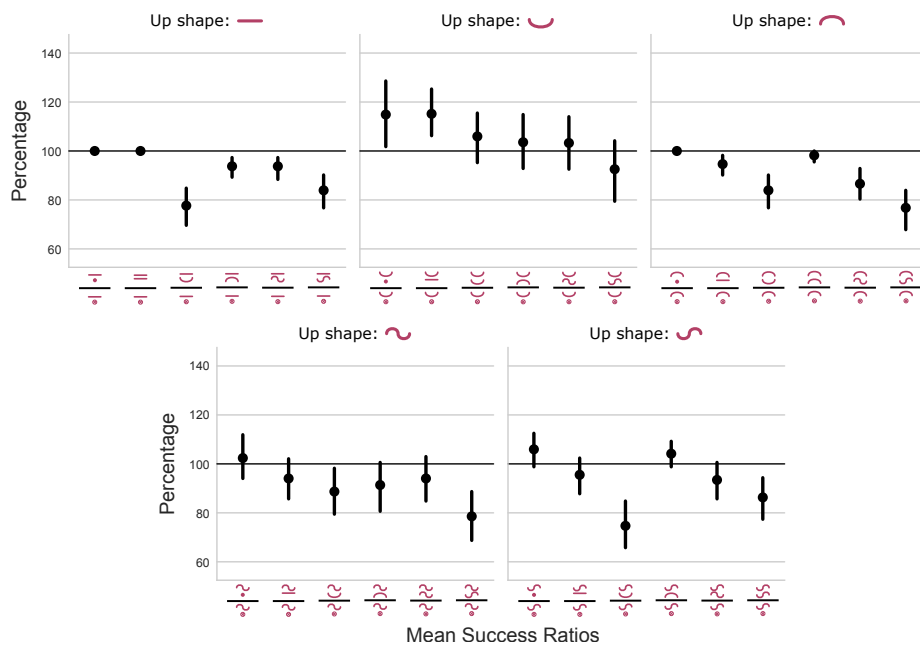


Figure 5: Mean success ratios and 95% CI for all the different combinations of *Strokes*. Ratios correspond to the mean of: success rates for a given combination divided by the mean success rate of the corresponding baseline.

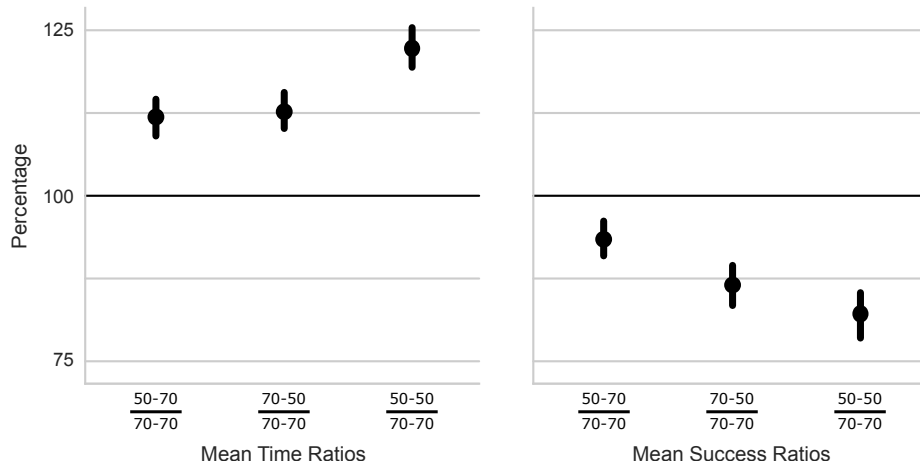


Figure 6: Mean time (resp. success) ratios and 95% CI for all the different combinations of Widths. Ratios correspond to the mean of: completion time (resp. success rates) for a given combination divided by the mean completion time (resp. success rate) of the combination with the larger Width.

320 doubles or more for all conditions except for *Downs* with \bullet . The greatest increases come from introducing curvy *Downs* when the forefinger follows a — path (Figure 4 leftmost graph); however, time values increase to level similar to combinations involving a down — .

5.5.2. Concurrency

325 In order to investigate whether a 2CSG system that can work on any touch devices without requiring special hardware, we first need to study whether two strokes can be performed simultaneously. We therefore need a metric to measure “how much” concurrent two strokes are. We use a tailored metric, inspired from the M-metric presented by Masliah and Milgram [39], to quantify users concurrency between the up and down strokes. It computes the difference of progression of each finger through
 330 time. We call this new metric the C-metric, for ‘Concurrency’ metric. The goal of our C-metric is to output a normalized value quantifying how much ‘concurrently’ strokes have been performed.

The metric is based on the drawing progression of each stroke at any given time of

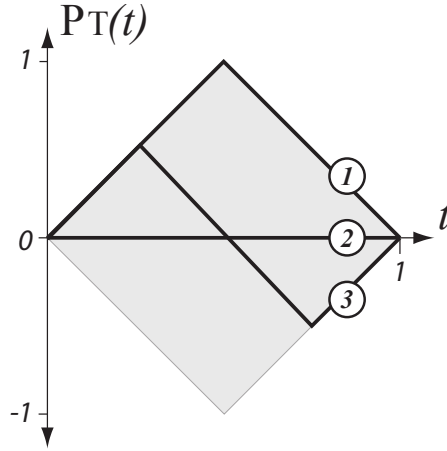


Figure 7: Three theoretical examples of trial completions: 1- Up stroke has been completely drawing before the down one; 2- Both strokes have been drawn concurrently all along; 3- A 3 phases completion: half of the up stroke alone, the full down stroke, and second half of the up stroke. The grey diamond shows the convex hull of PT , *i.e.* no trial could give a curve out of the diamond. Note: scales of the x- and y-axis are different.

the trial: $L_s(t)$, where $s \in \{\text{up, down}\}$, and $t \in [0, 1]$.

A stroke progression value, P , is therefore:

$$P_s(t) = \frac{L_s(t)}{L_s(1)} \quad (1)$$

³³⁵ P_s is normalized: at the beginning of the trial $P_s(0) = 0$, and at the end $P_s(1) = 1$. We can therefore compute PT , a function that gives the completion of a trial at a given time by:

$$PT(t) = P_{\text{up}}(t) - P_{\text{down}}(t) \quad (2)$$

Figure 7 shows examples of PT for three theoretical trials:

1. the curve with a perfect isosceles shape refers to a trial in which the up stroke has
³⁴⁰ been completely drawn before the down one;
2. the horizontal line refers to a trial in which both strokes have been done concurrently all along;









3. the curve with a ‘squared wave’ shape refers to a trial with the following sequence:
 half of the up stroke completed first, the down stroke completed in one go, and
 the second half of the up stroke was completed.

Figure 8 shows the mean PT of all the trials of the experiment, for each user and each combination. One can observe that most of the curves are skewed upwards, which means that most of the trials started moving the forefinger first.

We defined a concurrency metric C which corresponds to the normalized length of PT from 0 to 1. Considering the worst case, *i.e.* concurrency is never used (*e.g.* both curves 1 and 3), the total length of the PT curve is: $\max(\text{PT}) = 2\sqrt{0.5^2 + 1^2}$. In the best case, *i.e.* concurrency is used from start to finish (*e.g.* curve 2), $\min(\text{PT}) = 1$. The concurrency metric C is therefore:

$$C = 1 - \frac{\text{Len}(\text{PT}) - \min(\text{PT})}{\max(\text{PT}) - \min(\text{PT})} = 1 - \frac{\text{Len}(\text{PT}) - 1}{2\sqrt{1.25} - 1} \quad (3)$$

where $C \in [0, 1]$. $C = 0$ means no concurrency at all while $C = 1$ means the two strokes has been drawn simultaneously, at the same progression speed, all along the trial.

Figure 9 shows the distributions and estimated density of the C -metric across the different combinations. One can see a clear tendency to complete paths simultaneously when both strokes are identical (**parallel** strategy): distributions on the diagonal are skewed leftwards. For any other combinations, distributions are more spread out but participants still seem to prefer performing strokes somewhat concurrently. Only a few chose sequential strategy. For some combinations, *e.g.*  or , the two strategies are clearly identifiable and in the group of participants that used the sequential strategy their movements were almost never concurrent (C very close to 0). Another observation we can make is that sequential strategy is more used with , , , ,  and , when fingers need to move away from one another at the beginning of the movement.

5.6. Concurrency with Heat-Maps

In order to highlight the behaviour of the fingers in sequential and parallel strategies, we computed heat-maps of the contact positions on the screen during all trials (Figure 10).



Figure 8: Mean progression (*i.e.* PT) of each user during trials across the different combinations. The bold blue line represents the mean progression across participants and the light blue area, the 95% CI.



Figure 9: Distributions of the C -metric across the different combinations of strokes. Each color represents a single user. The blue line represents the estimated density function.

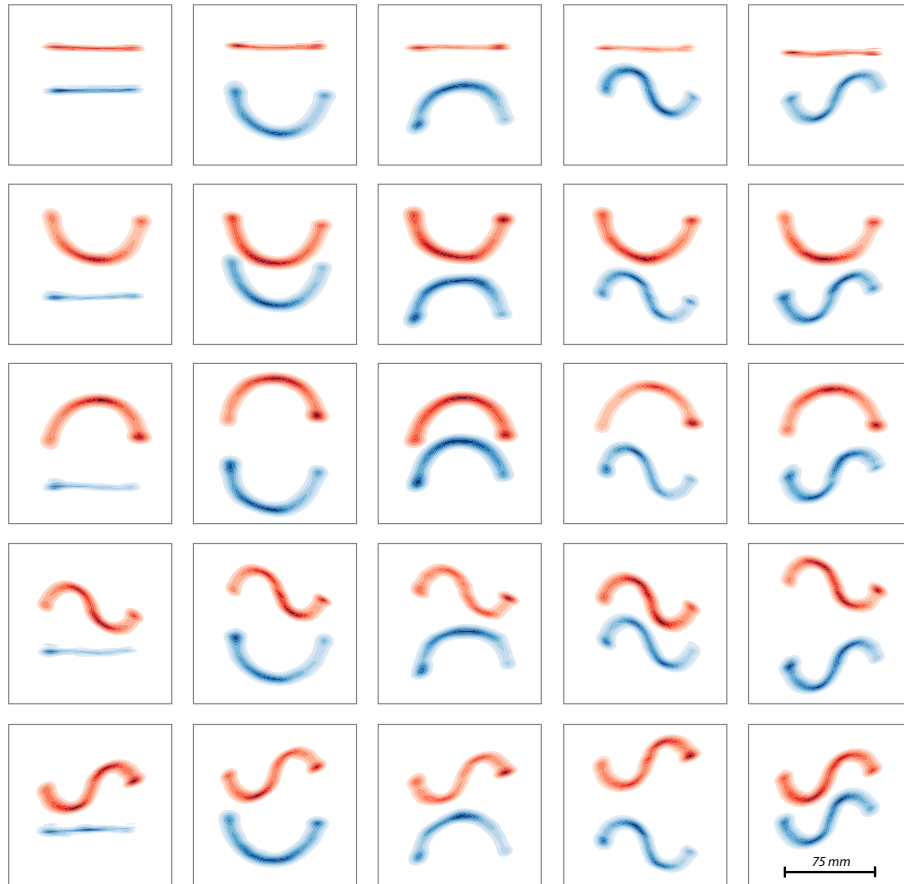


Figure 10: Heat-map of all touch events logged during study 1. Red and blue colors correspond to the *UpStrokes* and *DownStrokes* respectively. Each shade represents iso-proportions of the density in increment of 10% (from the lightest shades to the darkest).

These positions were extracted from all the touch events logged during the experiments. Each touch event contained both the forefinger and thumb positions. Hot-spots (*i.e.* darker colored areas) show areas of high event density. In the diagonal, corresponding to the **parallel** family, hotspots tends to be spread out around the apexes (*i.e.* mid-points of arcs). Given the concurrency metric results indicating the use of concurrent strategies, users most likely slowed down when reaching apexes. In the remaining families, hot spots are sharper and localized differently. Given the more mitigated concurrency metric results, we hypothesize that these hotspots most likely corresponds to stopping points used for sequential drawings of the strokes. They appear in the extremities (*e.g.* 3rd column and 4th row in figure 10) indicating that strokes were started and ended at different times, and the apexes (*e.g.* 5th column and 2nd row in figure 10) indicating that difficulties normally requiring careful tracing becomes too difficult when shapes are mixed. It concurs with our previous observations: 2CSG performance is linked to the number of turns and figure symmetry.

5.7. Qualitative results

Because of personal timing constraints (not due to a specific behaviour during the experiment), 3 participants were not able to commit to the debriefing session. Therefore this section presents the results of 11 of the 14 participants (mean age 27.54 years, SD=4.22 years; no left-handed; 2 females).

5.7.1. Fatigue and discomfort

7 participants perceived a little fatigue during the experiment. 4 perceived it on the thumb, from a slight discomfort to a feeling of an onset of tendinitis. The 3 others felt a global fatigue (arm, eyes, shoulder), and one of them mentioned that this fatigue could have influenced its strategy: independently from the figures, his/her gestures were more sequential for the last trials (developed in the following sub-section).

3 participants have been disturbed by a third finger (ring or little of the interacting hand) that touched the tablet during the steering interaction. This case is managed by the software in a similar way than when one finger leaves the tablet while the gesture is not complete: the software stops the trial, forcing the participant to re-start the trial

400 immediately. However, this issue mainly appeared in the first trials, and participants progressively corrected their hand position.






Since all participants were right-handed and were asked to draw the figures from left to right, they could have been disturbed by their hand occluding the figure, especially during the beginning of the gesture. However, only one participant mentioned it during
405 the debriefing.

5.7.2. *Strategies*

Without any influence from the experimenter, all participants thought about strategies to optimize their efficiency. During the interviews, we learned that they considered concurrency and sequentiality. 1 participant used sequential drawings only. Another
410 participant used concurrent drawings until (s)he felt a global fatigue and therefore switch to sequential only. For the other 9 participants, the method depended on the figure complexity, and more precisely on the eye movements imposed by the figures. Indeed, they used the concurrency by default, until the figure complexity forced the eyes to jump too frequently between the two strokes. At this level of complexity, they felt it would
415 be more efficient and relaxing to decrease the eye-jump frequency by making longer drawings one finger at a time.

5.7.3. *Perceived complexity*

First, only 4 participants felt impacted by the tunnel width, and 5 of the others explicitly said they didn't pay attention to it. This could suggest that either both widths
420 were wide enough to not pose problems, or that the following the paths was more challenging than meeting the required precision.

During the interview, we ask participants about combinations complexity. Without any suggestion from the interviewer, they spontaneously created families of complexity, and most of the participants made the same families as we mention above. As expected,
425 all participants highlighted the “one-finger group”, *i.e.* the **baseline** family as the easiest combinations to perform. In this group, complexity increases with the number of turns, *i.e.* the  is the easiest, followed by  and . Then  and  are the hardest.

6 participants mentioned the “fixed-thumb group”, *i.e.* the **anchored** family, and felt

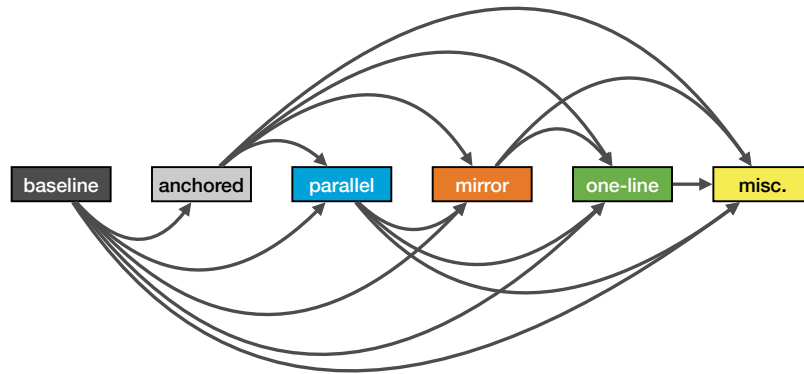


Figure 11: The majority graph from the classification of the 6 categories. Arrows mean ‘classified more times easier’, and the vertical ordering gives the final classification: baseline is the easier, and misc. the harder.

it “just a bit harder than the ‘one-stroke group’ (*i.e.* the **one-line** family). 1 participant
 430 felt a discomfort in his/her wrist and said that “fixing the thumb twists the hand”.

All participants proposed a family of “identical-strokes”, *i.e.* the **parallel** family. 9 participants ranked it just after **baseline** and **anchored** in terms of complexity. They expressed that, thanks to the constant distance between the fingers, this kind of combination is closed to the **baseline**.

435 Another family has been proposed by 9 participants, and ranked fourth. They call it the “symmetrical group”, *i.e.* our **mirror** family, and rank it after **parallel**.

All participants said that the hardest combinations are the one with **no straight line, no identical strokes, and no symmetry**, *i.e.* our **misc.** family Here all the participants had to use the sequential drawing method. According to 3 participants, this could be due
 440 to the difference between the number of inflection points in the up and down strokes.

These groupings suggests that participants perceived a similar complexity for the remaining gestures, *i.e.* the **one-line** family. Considering the interviews, this family should be classified in between **mirror** and **misc.**

Finally, even if not all participants mentioned all 6 families, they all used a classification of 3 to 6 elements. Using their classifications, we created a majority graph [40].
445 In this graph, showed in Figure 11, each node is one of the 6 families, and an arrow from A to B means that ‘A has been classified easier than B more times than B has been classified easier than A’. Applying the Copeland rules [41, 42], which consists in counting the number of out-coming arcs per node, we computed a global classification
450 represented by the horizontal position of the nodes (from the easiest, *i.e.* leftmost node, to the hardest, *i.e.* rightmost node).

The complexity classification therefore is (from the easiest family to the hardest one): **baseline**, **anchored**, **parallel**, **mirror**, **one-line** and **misc**. In order to corroborate these observations, we computed the mean time, mean *C*-metric and mean error rate for
455 each family. Results are shown in Figure 12.

The three figures present the same tendency, and corroborate the hypothesis for 5 of the 6 families. Unexpectedly, **mirror** has been overestimated, especially for time and error rate. According to the quantitative results, it places penultimate, close to **misc**. At first glance, one could consider the participants had been biased by symmetry, which
460 gives an illusion of simplicity to the figure. However, considering the *C*-metric indicator in Figure 12, and considering that complexity implies sequential drawings, the **mirror** group seems correctly placed on the majority graph of Figure 11.

This confirms that participants made a direct link between complexity and concurrency. For the 9 participants that made a link between drawing strategy and complexity,
465 all confirm they used concurrent drawing with **parallel**. 4 of them still used it with **mirror**, and 2 others with **one-line**. All participants used sequential drawing with **misc**.

6. Study 2: Recognition

In this second study we aim at evaluating how 2CSG could be used for command activation, and more precisely: how complexity affects performance, recognition and
470 learnability.

We used the same set of 35 combinations, and participants had to complete them with the forefinger and the thumb of their dominant hand. In order to force a command-

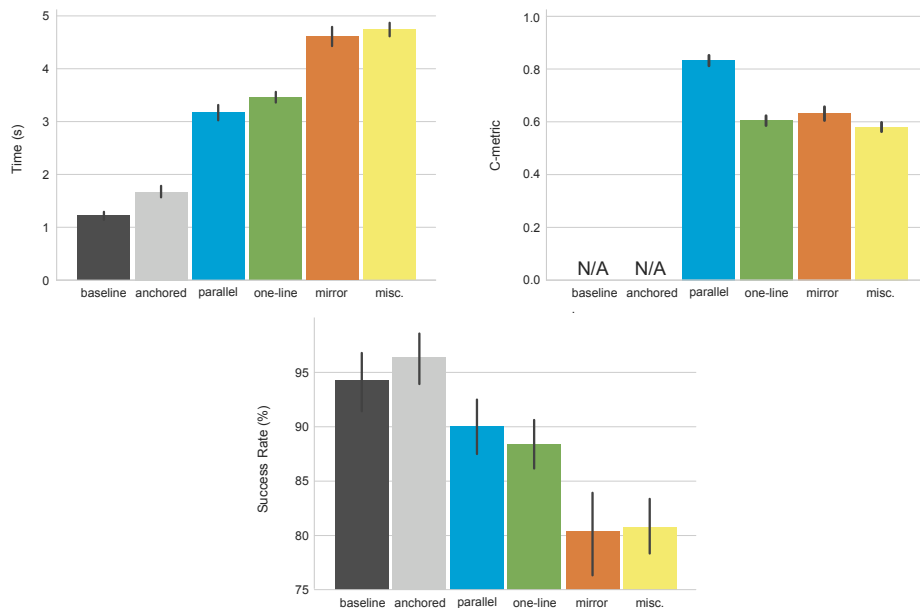


Figure 12: (left) Mean movement times and 95% CI for each family, classified from left to right by increasing perceived complexity. ‘mirror’ seems to be overestimated by the participants. (center) Mean C-metrics and 95% CI for each family, classified from left to right by increasing perceived complexity. There seems to be a correlation between complexity and concurrency. (right) Mean Error Rates and 95% CI for each family, classified from left to right by increasing perceived complexity. Here also, ‘mirror’ seems to be overestimated by the participants.

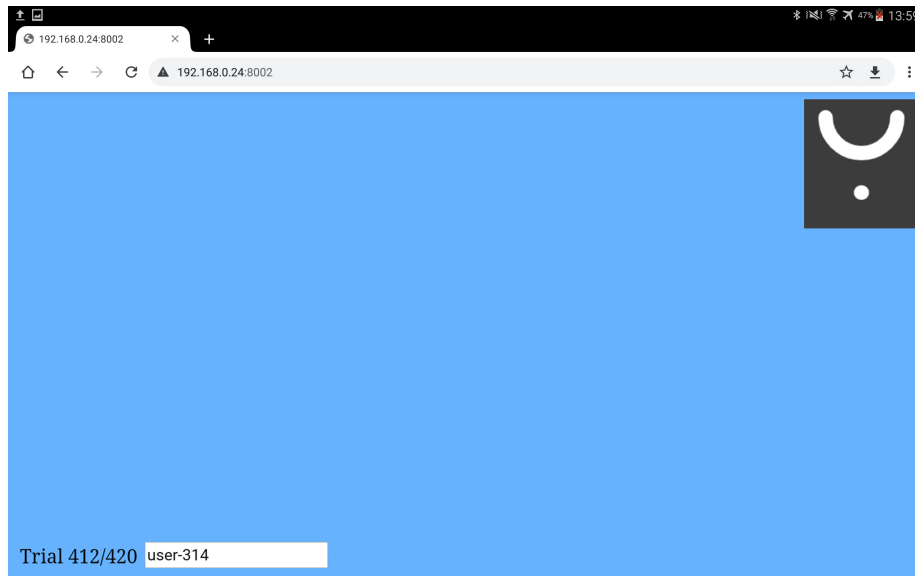


Figure 13: Interface of the second study. Participants can use all of the blue area to draw the top-right figure. Progression and participant's id are displayed on bottom-left.

like gesture, each trial needed to be done in less than two seconds. This time limit aimed at pressuring users just enough. It was empirically determined: one second was too fast, and pilot users did not even start trying; three seconds was too slow, and pilot users started slowing down and decomposing their gestures, which let them conscientize their movement. Exceeding the time cancelled the trial which had then to be repeated.

Figure 13 shows the interface of the experiment. Each combination to complete was successively displayed as a thumbnail on the top right corner of the screen. 2CSG could be performed anywhere on the screen. There were no starting, ending areas, nor visual feedback of the gesture during or after completion. The blue background turned green when the trial was completed in the time allowed, and turned red if time limit was exceeded, letting the participant know that the trial had to be done again.

We decided a display sequence of the combinations that consisted of 3 phases:

1. All the 35 combinations were randomly displayed once. If the time limit was exceeded, the combination was queued and prompted again later to ensure each trial was completed once.

2. Each combination was displayed again to be repeated 10 times in a row. The combinations order was the same than in phase 1 (taking into account reordering due to timing issue). If the time limit was exceeded, the same combination was proposed straight away until reaching 10 successful repetitions.
3. Identical to phase 1.

This specific ordering aimed at observing different facets in the learning process of 2CSG. It allowed to distinguish four kinds of gesture perception by participants:

- **New**: a combination a participant sees for the first time, *i.e.* all the trials of phase 1,
- **Known**: a combination a participant already encountered once, *i.e.* the first trial of each combination in phase 2),
- **Trained**: a combination participants trained on, *i.e.* the last trial of each combination in phase 2, after 9 repetitions,
- **Familiar**: a combination participants trained on previously, *i.e.* all the trials of phase 3.

6.1. Procedure

The experimental procedure was identical to the previous study. The experimenter started with a short demographic survey, showed next the interface and several 2CSG and finally gave the same instructions except:

- As there is no starting, ending area, nor tunnels, a trial could start and finish anywhere on the device screen. The size of the performed combination was up to the participant, and limited by the screen size only.
- The recording started when both the forefinger and thumb were in contact with the screen (or only the forefinger for \otimes). It ended when one finger was lift up.
- There was no training session, and it was not allowed to train on the table, nor any other surface to ensure that phase 1 trials were the first attempts.

There was no constraint on the tablet orientation, but all participants needed to complete the *UpStrokes* with their forefingers and the *DownStrokes* with their thumb along the horizontal axis of the tablet, to ensure rotation-invariance.

When all the trials were completed, the experimenter conducted a debriefing session, in which he asked about the strategies developed as well as the difficulties encountered depending on the combinations that were presented. In total, the experiment lasted from
520 30 to 45 minutes.

6.2. Experimental Design

The experiment used a $5 \text{ Ups} \times 7 \text{ Downs} \times 2 \text{ Repetitions}$ (phase 1 and 3) + $5 \text{ Ups} \times 7 \text{ Downs} \times 10 \text{ Repetitions}$ (phase 2) within-subject design. A total of 6300 trials were logged across all participants.

525 6.3. Participants and Apparatus

We recruited 15 new participants (mean age 30, SD=6.3 years; 2 left-handed; 8 females) that did not participate in study 1. All but one participant owned a smartphone and used it every day. Only 7 participants owned a tablet and used it every day. Participants barely used two-finger interactions in their daily life, apart for RST gestures
530 in map and picture applications.

As in the previous experiment, the study was conducted on a 1st generation 11 inches iPad pro (Retina display: 2388×1668 px resolution at 264ppi). The software was also written in HTML/CSS/JavaScript, and logged events through a web-socket sent to a python-written server running on a MacBook Pro 13inch fifth generation (Intel Core i5).

535 6.4. Analysis procedure

In the following we describe the two metrics we designed to explore recognition efficiency of 2CSG compared to single stroke gestures.

6.4.1. Recognition Score (Reco)

To assess gesture recognition rates in our study, we used state-of-the-art algorithms.
540 In general there are two types of recognizers families: state machines, suited to recognize continuous gestures, such as Proton [43, 44]; or matching algorithms, which compare strokes to templates such as the $\$$ -family [45, 46, 47, 48, 49]. Our goal was to compare each individual strokes to an *ideal* stroke, without taking into account the relative positioning, scale, and orientation of the two strokes composing a 2CSG. We, therefore,

545 decided to use the simplest, transparent and un-specific algorithm: \$1 recognizer [45].
It will act as a “metric” of “how close a drawn stroke is from its ideal counterpart”.

To allow \$1 to compute a score for a performed stroke, we first registered 5 theoretical strokes: —, ∪, ∩, ∞ and ∞, by using lists of 40 equally spreadout points that define the ideal drawings we showed to the participants during the experiment. Then,
550 for each trial, we run \$1 on each *Up* and *Down Stroke* of the combination. \$1 compares a stroke to its theoretical counterpart, and gives a score, between 0 and 1, where 1 is a perfect match. The recognition score of a trial, **Reco**, is an average of the \$1 score of each stroke of its combination:

$$\text{Reco}\left(\begin{matrix} S_u \\ S_d \end{matrix}\right) = \frac{\text{Score}(S_u) + \text{Score}(S_d)}{2} \quad (4)$$

555 To be noted, for the **baseline** family, $\text{Reco}\left(\begin{matrix} S_u \\ \otimes \end{matrix}\right)$ is equal to $\text{Score}(S_u)$

As \$1 is not able to detect a dot as a stroke, we defined a recognition score for the

- . It uses the width of the *DownStroke* and the width of the *UpStroke*:

$$\text{Score}(\bullet) = \max\left(0, 1 - \frac{\text{width}_{\text{down}}}{\text{width}_{\text{up}}}\right) \quad (5)$$

6.4.2. Score of Similarity (*Sim*)

In order to quantify how *DownStrokes* influence the recognition of *UpStrokes*, we compute a score of similarity (**Sim**). For a trial, this score is the ratio of $\text{Score}(S_u)$ with the mean **Reco** of the corresponding baseline:

$$\text{Sim}\left(\begin{matrix} S_u \\ S_d \end{matrix}\right) = \frac{\text{Score}(S_u)}{\text{Mean Reco}\left(\begin{matrix} S_u \\ \otimes \end{matrix}\right)} \quad (6)$$

560

6.5. Quantitative results

6.5.1. Recognition Score (*Reco*)

In Figure 14-left, cells show the **Reco** score of each combinations. The top row corresponds to the **baseline** family. To confirm that the complexity of our set of strokes

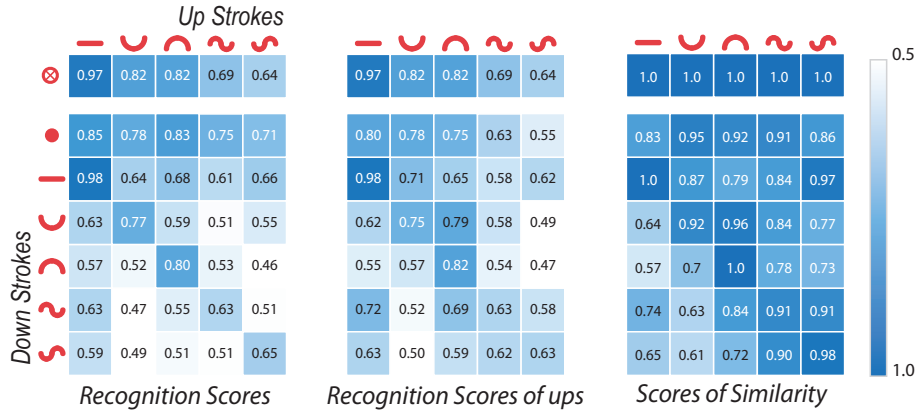


Figure 14: (left) Mean Recognition Scores (i.e. **Reco) for all the different combinations of Strokes. (center) Mean Recognition Scores of the *UpStrokes* only (i.e. **Score(S_u)**) for all the different combinations of Strokes. (right) Mean Scores of Similarity (i.e. **Sim**) for all the different combinations of Strokes.**

565 varies with radius of curvatures and inflection points, we ran a one-way ANOVA. It shows a significant effect of stroke on **Reco** ($F_{4,11} = 127, p < .001$). Post-hoc Tukey tests, reveals differences between \otimes and the the rest (all $p < .001$), as well as between the group [\cup, \cap] and the group [$\infty, \infty\bullet$] (all $p < .001$). As expected, it seems that **Reco** decreases when curvature and inflection points increase.

570 A one-way ANOVA showed significant effect of *DownStroke* on **Reco** ($F_{6,9} = 186, p < .001, \eta_p^2 = .16$ large effect size). Post-hoc Tukey tests, reveals:

- no difference between \otimes and \bullet ,
- differences between the group [\otimes, \bullet] and the rest (all $p < .001$),
- differences between — and the rest (all $p < .001$),
- 575 • differences between \cup and [$\infty, \infty\bullet$] (both $p < .001$)

A one-way ANOVA showed significant effect of 2CSG family on the recognition scores ($F_{5,10} = 340, p < .001, \eta_p^2 = 0.23$ large effect size). Post-hoc Tukey tests, reveals:

- no difference between **baseline**, **anchored** and **parallel**,
- differences between the group [**baseline**, **anchored**, **parallel**] and the rest (all





580

$p < .001$),

- differences between **one-line** and the rest (all $p < .001$)

These trends can be observed in Figure 14-left:



585

- we observe that **baseline** (first row), **anchored** (second row) and **parallel** (diagonal from  to ) obtain better mean scores than the remaining groups.
- as expected, lowest scores are for the **misc.** family, especially when combining strokes with a different number of inflection points. For example,  and  had the worst scores (0.47 and 0.46 resp.).










6.5.2. Score of Similarity (Sim)

590



In Figure 14-right, cells show the mean scores of similarity (Sim) for each combinations.

Overall, as expected adding *DownStrokes* lower the *UpStroke* recognition score. The more complex the *DownStroke* the lower the score is. However simple *DownStrokes* such as  and  do not affect the recognition score drastically which may be leveraged as ways to modify command gesture sets.



595

Figure 14-right illustrates relatively unexpected results on how *DownStrokes* affect *UpStrokes*. First, considering the lower-left corner, and the color gradient from left to right in general, one can see that the more complex a *Up* stroke is the less its score is affected by the *DownStrokes*. For instance, comparing  and , one can see that  affects a lot more  (56.7%) than  (73.4%). Also,  is the most affected stroke in general. This result can be explained by the extreme simplicity of ,  and : the simpler a stroke is, the more a small variation can affect the recognizer.

600

Next, the second line of the matrix shows the influence of . While using  only consists of drawing single strokes with the forefinger while the thumb is fixed on the tablet, it still influences the *UpStrokes*, and similarities vary from 82.5% to 95.1%.

605

Finally, while we expected making the same stroke with the two fingers should not influence much the \$1 score of *Ups*, the diagonal of the matrix shows that the difference with a single stroke can be up to 8.7% (for  and ) .

6.5.3. 2CSG Families Analysis

After grouping shape combinations based on 2CSG families, we look at the completion time, **Reco**, and **Sim**. We focus the analysis on two factors: *Families* and *Repetitions* of the combination. Levels for the *Repetitions* factor are 1 (trial of phase 1), 2 to 11 (trials of phase 2) and 12 (trial of phase 3).

Mean completion times by *Families* are: **baseline** 523ms (SD=257ms), **anchored** 757ms (SD=283ms), **parallel** 670ms (SD=319ms), **one-line** 1040ms (SD=354ms), **mirror** 1175ms (SD=346ms), **misc.** 1320ms (SD=387ms).

Mean completion times by *Repetitions* are: **1** 1159ms (SD=468ms), **2** 976ms (SD=448ms), **3** 959ms (SD=454ms), **4** 939ms (SD=435ms), **5** 945ms (SD=437ms), **6** 943ms (SD=432ms), **7** 928ms (SD=424ms), **8** 928ms (SD=428ms), **9** 915ms (SD=416ms), **10** 912ms (SD=427ms), **11** 877ms (SD=410ms), **12** 941ms (SD=239ms).

A one-way ANOVA showed significant main effect of *Family* ($F_{5,10} = 888, p < .001, \eta_p^2 = 0.4$ medium effect size) and *Repetition* ($F_{11,4} = 18, p < .001, \eta_p^2 = 0.02$ barely no effect size) on the completion time. However, there was no interaction between *Family* and *Repetition*. Post-hoc Tukey tests, reveals differences between all *Family* (all $p < .001$). As for the *Repetition*, the first repetition was significantly slower than the other (all $p < .001$), but no further difference were found.

Mean **Reco** by *Families* are: **baseline** 0.79 (SD=0.21), **anchored** 0.79 (SD=0.16), **parallel** 0.76 (SD=0.21), **one-line** 0.63 (SD=0.25), **mirror** 0.53 (SD=0.22), **misc.** 0.51 (SD=0.22).

A one-way ANOVA showed significant main effect of *Family* ($F_{5,10} = 339, p < .001, \eta_p^2 = 0.2$ small effect size) on **Reco**. However, there was no main effect of *Repetition* nor interaction between *Family* and *Repetition*. Post-hoc Tukey tests, reveals differences between **baseline** and [**one-line, mirror, misc.**] (all $p < .001$); between **anchored** and [**one-line, mirror, misc.**] (all $p < .001$); between **parallel** and [**one-line, mirror, misc.**] (all $p < .001$); between **mirror** and **one-line** ($p < .001$) and between **one-line** and **misc.** ($p < .001$). These results follow nicely participants grouping

Mean **Sim** by *Families* are: **baseline** 0.79 (SD=0.21), **anchored** 0.70 (SD=0.28), **parallel** 0.76 (SD=0.24), **one-line** 0.63 (SD=0.36), **mirror** 0.64 (SD=0.29), **misc.** 0.55

(SD=0.30).

A one-way ANOVA showed significant main effect of *Family* ($F_{5,10} = 99, p < .001$,
640 $\eta_p^2 = 0.07$ barely no effect size) on **Sim**. However, there was no main effect of *Repetition*
nor interaction between *Family* and *Repetition*. Post-hoc Tukey tests, reveals differences
between all *Families* (all $p < .001$) but **one-line** and **mirror**, and **baseline** and **parallel**.

There is no evidence of improvement nor degradation of the quality of the com-
binations throughout the experiment. However, there is still a decreasing trend when
645 looking at the time. It seems that participants slightly improve in terms of completion
time without affecting their drawing quality.

Given the simplicity of the shapes, and even though no feedback on successes nor
failures were given, we hypothesized participants to be able to self determine whether
their gestures were successful (motor-wise) or not compared to what they mentally
650 pictured. We were therefore expected improvements over time thanks to participants
self-correcting their gestures during the successive repetitions. However, we observed
no differences between the four learning levels, *i.e.* **New** (or repetition 1), **Known**
(repetition 2), **Trained** (repetition 11) and **Familiar** (repetition 12), except between
New and the rest.

655 6.6. Qualitative results

6.6.1. Fatigue

11 participants felt some physical or mental fatigue. About physical fatigue, 3
participants felt it on their wrist, 2 on their thumb, and 1 on the hand as a whole. This
fatigue was mainly caused by the high number of *Repetitions*, especially for participants
660 who regularly exceeded the time limit, and therefore redid a higher number of trials.
3 participants expressed a wrist discomfort with **anchored**. Similarly, 2 participants
expressed a thumb fatigue.

However, considering the high number of repetitions as not ecological, and con-
sidering that most of the participants expressed fatigue as “light tensions in the hand”,
665 making command activation with 2CSG does not seem physically demanding.


About mental fatigue, 5 participants complained, and correlated it to the experiment
itself, *i.e.* the number of repetitions. However, 11 participants complained about the

stress caused by **misc.**. One participant said “this gestures are nearly impossible to make!”.

670 6.6.2. *Strategies*

The nature of the trials do not let much room for strategy. Only four participants used sequential movements, and this was for **misc.** only. Indeed, as mentioned by one participant, using sequentiality takes more time and is not consistent with the command gesture principle, neither with the 2s time limit. Another participant started
675 the experiment with sequential movements, but quickly switched to parallelism. As they could rest before each trial, two participants mentally pictured the gestures before each trials, especially in phase 1 and 3.

6.6.3. *Perceived complexity*

As expected, the same complexity groups (our 6 families) mentioned in study 1
680 have also been mentioned by the participants of study 2. All the 6 have been mentioned by 7 participants, 5 participants simply did not mentioned **one-line**. As expected too, **baseline** is always the easiest and **misc.** the hardest. 13 participants ranked **parallel** or **anchored** slightly harder than **baseline**. The two remaining participants stated that **anchored** was not that easy because it forces the wrist to make an arc. All
685 the 7 participants that mentioned the 6 families placed **one-line** before last in their complexity scale. They mainly did so because of the frustration involved by the influence of the curved-stroke on the line-stroke. It induced feelings of *lack of precision*, and *fear that the gesture will not be recognized*. Among all the participants, 6 mentioned the influence of strokes with a higher number of turns over the ones with a lower number
690 of turns. The typical example was  where participants were convinced that a wave should appear in the resulting *DownStroke*.

As in study 1, we computed a majority graph, and obtained the same result illustrated in Figure 11.

6.6.4. *Learning*

695 8 participants felt they were learning across phase 2. One participant said that this phase seemed to be beneficial for all the gestures, but all said it improved performance

on medium and highly demanding gestures mainly. Also, two participants had the feeling that this learning was not “very” sustainable, and were not sure to perform trials of phase 3 as well in as the last repetition of phase 2.

700 **7. Discussion**

Results from our experiments let us foresee how stroke complexity in 2CSG affects performance, and especially how thumb movements could be combined with concurrent forefinger strokes. In the following, we identify the takeaways from both experiments. Based on these takeaways, we propose a set of guidelines for future use of 2CSG.





705 Even though the 30 stroke combinations we studied cannot be exhaustive, they still provide a decent overview on how complexity affects performance both when precision is needed and when discrete commands are issued.

To ease the generalizability of our results, we grouped the stroke combinations into different families. It is to be noted that these families were spontaneously retrieved
710 by participants when describing stroke combinations during the interview phase. This concurring grouping acts as a sanity check, therefore providing us with a sensible lens through which analyzing the results.

Finally, we opted for open-ended questions on fatigue and learning to not emphasize anything in particular and let participants chose their own level of details at which they
715 wanted to discuss. Moreover, families were not introduced to the participants, since we wanted them to group the gestures. Therefore to not influence them, we would have had to compare all pairs of combination making the experiment potentially too long. The open-ended form appeared to be the best compromise.

7.1. Family Performance

720 In both experiments, the **anchored** family yields performance that are close to the **baseline** (confirming **H4**). Adding a static thumb while performing strokes with the forefinger therefore seems to be a usable alternative. On the contrary all the other thumb gestures we studied did affect negatively performance at various degrees (partially confirming **H1**).

725 In our study focusing on precision, no other families did perform better than simply
sequentially drawing strokes. While not necessarily a surprise for most of them, we
did not expect this result for the **one-line** family (e.g. , ,  and ): drawing
time increased by more than twice. Similarly, and even more unexpected, the **parallel**
gestures have significantly worse performance than their **baseline** counterparts (invali-
730 dating **H2**). We hypothesized that to perform **parallel** gestures participant would fix the
distance between their thumb and their index and only focus on the position of either
one of these fingers. We thought that strategy to be relatively close to performing a
baseline gesture. However, this strategy may have been too simplistic and overlooked
the biomechanic of the hand and the angle at which participants drew their gestures
735 (which was not controlled): maintaining a fixed distance between the thumb and the
index may not be a straightforward task. As a note, results shown in figure 9 tend to
partially invalidate **H3**: only strokes from the **parallel** family tend to be systematically
drawn concurrently.

740 In our study focusing on gesture recognition, we identified more families that could
be usable alternatives. While we expected to have higher recognition rates for each
gestures of the **baseline**, we only focus on the relative changes in the recognition score
as one could argue that a better recognition algorithm and more samples in the training
set could be added. Therefore, considering the Similarity score of the families we tested,
745 not only the **anchored** family (confirming **H4** again) but the **one-line** and **parallel** ones
seem promising alternatives. Perhaps these families, hard to perform but recognizable,
could be used in non-frequent situations and/or critical commands (e.g., closing an
app) since they should not be triggered inadvertently. We think it opens up interesting
questions to explore in future iterations.


750

Both studies tend to show that the **parallel** family is more different from the **an-**
chored family than expected (thus invalidating **H5**), and (at least from study one) much
closer to the **one-line** family.

7.2. Guidelines

755 Results on the use of 2CSG are quite anticlimactic. On the one hand we identified families of gestures that can be used and could easily be integrated to augment the interaction bandwidth on touchscreens. On the other hand stroke complexity can not be easily leverage to drastically increase this bandwidth. In the following, we provide some guidelines and example scenarios on usable 2CSG families.

760 When precise forefinger drawing is needed, no precision with the thumb should be required, and only the **anchored** family should be leveraged. The thumb can be used as a modifier, but it should be interpreted as on or off the screen. For instance, in a selection tool, the thumb can act as a quasimode for multiselection: items that cross the forefinger path are selected only if the thumb is also in contact with the screen. To be noted, that in
765 our study we required precision for both the forefinger and the thumb. In future works, we should consider *unconstrained* thumb movements: while the forefinger draws with precision, can the thumb moves without constraints or need to remain static? Good performance with unconstrained thumb movements would allow the use of a modifier for longer gestures, and be used on bigger screens such as tabletops.

770 When issuing discrete command gestures, the **anchored**, **one-line** and **parallel** families can be leveraged. Simple **parallel** gestures have already been explored (*e.g.*, parallel lines in various direction for multitouch marking menu [33]) and even are in use in commercial devices (*e.g.*, accessing advanced settings through two-finger bezel swipes downwards on Android), but the use of more complex **parallel** 2CSG can be
775 added to the toolbox. To the best of our knowledge, the **anchored** (with forefinger gestures) and **one-line** families have yet to be used. A system could therefore: 1) use an **anchored** thumb as a quasimode to explicitly issue discrete command through gestures as commonly studied in the literature (*e.g.*, [28]) (for instance a  to turn the WiFi on/off); the **one-line** family could also extend this command interpretation quasimode;
780 2) use the **parallel** family as already in place (*e.g.*, [33] and Android settings example) but with a bigger set of gestures.

7.3. Future Work

A first extension of this work, could be to further explore the 2CSG design space in order to complete our guidelines, by studying new gestures such as inclined lines (*e.g.* < or > shapes), broken lines (*e.g.* W shapes), or even switching roles (*e.g.* anchoring the forefinger instead of the thumb).

Our second study included two left-handed persons. The handedness factor did not reveal any difference. However, given our sample size, this lack of difference does not mean there is none. Future work may consider investigating the effect of handedness, however we hypothesize that the main difference is mechanical (*e.g.*, participants either pull or push when sliding their fingers), and thus that such an experiment will not show any critical difference in performance, the performance bottleneck being arguably more due to cognitive rather than mechanical constraints.

In our second study, we used a \$1 recognizer on each stroke to analyze them independently. In future works, other strategies could be envisioned to increase the interaction bandwidth. A simple example, would be to use a \$N recognizer on the whole stroke combination to investigate the strokes relative distance as a new input parameter. One could thus envision, differentiating **anchored** gestures depending on how far a thumb is from the stroke. The same idea could be applied to **one-line** and **parallel** families. To go even further, other properties could also be studied (*e.g.*, relative ratios, x- and y-offsets). However, it is important to keep in mind that a potential trade-off between number of parameters and usability could arise.

Touch-device sizes give an inherent maximum distance between thumb and forefinger, which can limit mobility and precision for making a 2CSG, especially on small devices. Moreover, Chih-Chun et al. showed in 2016 that “7” display [...] underperformed the 9”-11” displays” on several tasks (like drag-and-drop), even though they had the same index of difficulty [50]. As 2CSG should therefore be smaller, slower and more complex to perform on small devices, one could envision that performance could be different. However, in terms of gesture ranking and guidelines, we do not expect a significant difference in general, maybe for specific fingers only that have restrained range of movements (*e.g.*, long nails).

We used \$1 recognizer for our purposes, but better recognition could be used for real-world system. However, our interest laid in the comparison of these scores. \$P,
815 \$P+, \$Q and protractor versions, and including context awareness (mobile, wearable, low-vision, ...) could lead to faster and/or better recognition results overall, but the use of a simple recognition algorithm helps us highlight the difference in drawing themselves compared to an ideal stroke.

This paper only focuses on the motor action as a first step. Our rational behind our
820 gesture set was to use a definition of drawing difficulty [34]. However, an elicitation study could be a complementary research, and might validate our choice of gesture or spring up new types of 2CSG, as well as scenarios and associated actions, therefore potentially leading to new guidelines. For instance, families that are hard to perform motor-wise may still be very much appreciated (e.g., direct link between the gesture
825 and the action, ...).

8. Conclusion

This paper presents two experiments that investigate **two concurrent stroke gestures** (2CSG) benefits and limitations. While we aimed at proposing a large amount of new efficient gestures for tactile devices, we observed that making concurrent strokes
830 with our thumb and forefinger is not as an easy task as expected. However, we still identified several families that are promising to possibly enhance precise drawing and/or command gestures. For precise drawings, **anchored** could be leveraged as a modifier. For command gestures, **anchored**, **one-line** and **parallel** could be leveraged as new quasimodes, and deserve to be studied further alongside additional gestures.

835 References

References

- [1] J. Wagner, E. Lecolinet, T. Selker, Multi-finger chords for hand-held tablets: Recognizable and memorable, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14, Association for Computing

- 840 Machinery, New York, NY, USA, 2014, p. 2883–2892. doi : 10.1145/2556288.2556958.
URL <https://doi.org/10.1145/2556288.2556958>
- [2] K. Seipp, K. Devlin, Landscape vs portrait mode: Which is faster to use on your smart phone?, in: Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '13, 845 Association for Computing Machinery, New York, NY, USA, 2013, pp. 534–539. doi : 10.1145/2493190.2494422.
URL <https://doi.org/10.1145/2493190.2494422>
- [3] R. Xiao, J. Schwarz, C. Harrison, Estimating 3d finger angle on commodity 850 touchscreens, in: Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces, ITS '15, Association for Computing Machinery, New York, NY, USA, 2015, pp. 47–50. doi : 10.1145/2817721.2817737.
URL <https://doi.org/10.1145/2817721.2817737>
- [4] A. Goguey, G. Casiez, D. Vogel, C. Gutwin, Characterizing Finger Pitch and Roll 855 Orientation During Atomic Touch Actions, Association for Computing Machinery, New York, NY, USA, 2018, pp. 1–12.
URL <https://doi.org/10.1145/3173574.3174163>
- [5] S. Feng, G. Wilson, A. Ng, S. Brewster, Investigating pressure-based interactions with mobile phones while walking and encumbered, in: Proceedings of the 17th 860 International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct, MobileHCI '15, Association for Computing Machinery, New York, NY, USA, 2015, pp. 854–861. doi : 10.1145/2786567.2793711.
URL <https://doi.org/10.1145/2786567.2793711>
- [6] A. Goguey, S. Malacria, C. Gutwin, Improving Discoverability and Expert Performance in Force-Sensitive Text Selection for Touch Devices with Mode Gauges, 865 Association for Computing Machinery, New York, NY, USA, 2018, pp. 1–12.
URL <https://doi.org/10.1145/3173574.3174051>

- [7] A. Goguey, D. Vogel, F. Chevalier, T. Pietrzak, N. Roussel, G. Casiez, Leveraging finger identification to integrate multi-touch command selection and parameter manipulation, *International Journal of Human-Computer Studies* 99 (2017) 21–36. doi:<https://doi.org/10.1016/j.ijhcs.2016.11.002>.
URL <https://www.sciencedirect.com/science/article/pii/S1071581916301537>
- [8] C. Harrison, J. Schwarz, S. E. Hudson, Tapsense: Enhancing finger interaction on touch surfaces, in: *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST '11*, Association for Computing Machinery, New York, NY, USA, 2011, pp. 627–636. doi:[10.1145/2047196.2047279](https://doi.org/10.1145/2047196.2047279).
URL <https://doi.org/10.1145/2047196.2047279>
- [9] P. Ewerling, A. Kulik, B. Froehlich, Finger and hand detection for multi-touch interfaces based on maximally stable extremal regions, in: *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces, ITS '12*, Association for Computing Machinery, New York, NY, USA, 2012, pp. 173–182. doi:[10.1145/2396636.2396663](https://doi.org/10.1145/2396636.2396663).
URL <https://doi.org/10.1145/2396636.2396663>
- [10] C. Appert, S. Zhai, Using strokes as command shortcuts: Cognitive benefits and toolkit support, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09*, Association for Computing Machinery, New York, NY, USA, 2009, pp. 2289–2298. doi:[10.1145/1518701.1519052](https://doi.org/10.1145/1518701.1519052).
URL <https://doi.org/10.1145/1518701.1519052>
- [11] Y. Li, Gesture search: A tool for fast mobile data access, in: *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology, UIST '10*, Association for Computing Machinery, New York, NY, USA, 2010, pp. 87–96. doi:[10.1145/1866029.1866044](https://doi.org/10.1145/1866029.1866044).
URL <https://doi.org/10.1145/1866029.1866044>
- [12] K. Hinckley, P. Baudisch, G. Ramos, F. Guimbretiere, Design and analysis of delimiters for selection-action pen gesture phrases in scriboli, in: *Proceedings*

of the SIGCHI Conference on Human Factors in Computing Systems, CHI '05, Association for Computing Machinery, New York, NY, USA, 2005, pp. 451–460. doi : 10.1145/1054972.1055035.

900 URL <https://doi.org/10.1145/1054972.1055035>

[13] D. Goldberg, C. Richardson, Touch-typing with a stylus, in: Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems, CHI '93, Association for Computing Machinery, New York, NY, USA, 1993, pp. 80–87. doi : 10.1145/169059.169093.

905 URL <https://doi.org/10.1145/169059.169093>

[14] G. Bailly, E. Lecolinet, Y. Guiard, Finger-count & radial-stroke shortcuts: 2 techniques for augmenting linear menus on multi-touch surfaces, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA, 2010, pp. 591–594.

910 URL <https://doi.org/10.1145/1753326.1753414>

[15] S. Knoedel, M. Hachet, Multi-touch rst in 2d and 3d spaces: Studying the impact of directness on user performance, in: 2011 IEEE Symposium on 3D User Interfaces (3DUI), 2011, pp. 75–78. doi : 10.1109/3DUI.2011.5759220.

[16] A. Goguey, M. Nancel, G. Casiez, D. Vogel, The performance and preference
915 of different fingers and chords for pointing, dragging, and object transformation, in: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, CHI '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 4250–4261. doi : 10.1145/2858036.2858194.

URL <https://doi.org/10.1145/2858036.2858194>

920 [17] C. Lang, M. Schieber, Human finger independence: limitations due to passive mechanical coupling versus active neuromuscular control., *Journal of neurophysiology* 92 5 (2004) 2802–10.

[18] R. J. K. Jacob, L. E. Sibert, D. C. McFarlane, M. P. Mullen, Integrality and separability of input devices, *ACM Trans. Comput.-Hum. Interact.* 1 (1) (1994)

- 925 3–26. doi:10.1145/174630.174631.
URL <https://doi.org/10.1145/174630.174631>
- [19] S. Handel, S. Imai, The free classification of analyzable and unanalyzable stimuli, *Attention, Perception, & Psychophysics* (1972).
- [20] M. W. Krueger, T. Gionfriddo, K. Hinrichsen, Videoplace—an artificial reality, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '85, Association for Computing Machinery, New York, NY, USA, 1985, pp. 35–40. doi:10.1145/317456.317463.
930 URL <https://doi.org/10.1145/317456.317463>
- [21] T. Moscovich, J. F. Hughes, Multi-finger cursor techniques, in: *Proceedings - Graphics Interface*, 2006.
935
- [22] T. Moscovich, J. F. Hughes, Indirect mappings of multi-touch input using one and two hands, in: *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*, ACM Press, New York, New York, USA, 2008, p. 1275. doi:10.1145/1357054.1357254.
940 URL <http://portal.acm.org/citation.cfm?doid=1357054.1357254>
- [23] M. A. Nacenta, P. Baudisch, H. Benko, A. Wilson, Separability of spatial manipulations in multi-touch interfaces, in: *Proceedings of Graphics Interface 2009, GI '09*, Canadian Information Processing Society, CAN, 2009, pp. 175–182.
- [24] V. Rusnák, C. Appert, O. Chapuis, E. Pietriga, Designing coherent gesture sets for multi-scale navigation on tabletops, in: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI '18*, Association for Computing Machinery, New York, NY, USA, 2018, pp. 1–12. doi:10.1145/3173574.3173716.
945 URL <https://doi.org/10.1145/3173574.3173716>
- [25] G. Kurtenbach, W. Buxton, User learning and performance with marking menus, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '94*, Association for Computing Machinery, New York, NY, USA,
950

1994, pp. 258–264. doi : 10.1145/191666.191759.

URL <https://doi.org/10.1145/191666.191759>

955 [26] R. Zeleznik, T. Miller, Fluid inking: augmenting the medium of free-form inking with gestures, in: Proceedings of Graphics Interface 2006, GI 2006, Canadian Human-Computer Communications Society, Toronto, Ontario, Canada, 2006, pp. 155–162.

[27] P. O. Kristensson, S. Zhai, Command strokes with and without preview: Using
960 pen gestures on keyboard for command selection, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07, Association for Computing Machinery, New York, NY, USA, 2007, pp. 1137–1146. doi : 10.1145/1240624.1240797.

URL <https://doi.org/10.1145/1240624.1240797>

965 [28] N. Banovic, F. C. Y. Li, D. Dearman, K. Yatani, K. N. Truong, Design of unimanual multi-finger pie menu interaction, in: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces, ITS '11, Association for Computing Machinery, New York, NY, USA, 2011, pp. 120–129. doi : 10.1145/2076354.2076378.

970 URL <https://doi.org/10.1145/2076354.2076378>

[29] E. Ghomi, S. Huot, O. Bau, M. Beaudouin-Lafon, W. E. Mackay, Arpège: Learning multitouch chord gestures vocabularies, in: Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces, ITS '13, Association for Computing Machinery, New York, NY, USA, 2013, pp. 209–218.
975 doi : 10.1145/2512349.2512795.

URL <https://doi.org/10.1145/2512349.2512795>

[30] Y. Luo, D. Vogel, Pin-and-cross: A unimanual multitouch technique combining static touches with crossing selection, in: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, UIST '15, Association for
980 Computing Machinery, New York, NY, USA, 2015, pp. 323–332. doi : 10.1145/

2807442.2807444.

URL <https://doi.org/10.1145/2807442.2807444>

- [31] Y. Rekik, R. daniel Vatavu, L. Grisoni, Understanding users'perceived difficulty of multi-touch gesture articulation, in: In ICMI'12, ACM, 2014.
- 985 [32] L. A. Leiva, D. Martín-Albo, R.-D. Vatavu, Gato: Predicting human performance with multistroke and multitouch gesture input, in: Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '18, Association for Computing Machinery, New York, NY, USA, 2018. doi : 10 . 1145/3229434 . 3229478.
- 990 URL <https://doi.org/10.1145/3229434.3229478>
- [33] G. J. Lepinski, T. Grossman, G. Fitzmaurice, The design and evaluation of multitouch marking menus, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '10, Association for Computing Machinery, New York, NY, USA, 2010, pp. 2233–2242. doi : 10 . 1145/1753326 . 1753663.
- 995 URL <https://doi.org/10.1145/1753326.1753663>
- [34] P. Viviani, C. Terzuolo, Trajectory determines movement dynamics, *Neuroscience* (1982). doi : 10 . 1016/0306 - 4522 (82)90277 - 9.
- [35] D. Freeman, H. Benko, M. R. Morris, D. Wigdor, Shadowguides: Visualizations for in-situ learning of multi-touch and whole-hand gestures, in: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces, ITS '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 165–172.
- 1000 doi : 10 . 1145/1731903 . 1731935.
- URL <https://doi.org/10.1145/1731903.1731935>
- [36] J. O. Wobbrock, M. R. Morris, A. D. Wilson, User-defined gestures for surface computing, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 1083–1092. doi : 10 . 1145/1518701 . 1518866.
- 1005 URL <https://doi.org/10.1145/1518701.1518866>

- [37] Y. Rekik, L. Grisoni, N. Roussel, Towards many gestures to one command: A user study for tabletops, in: P. Kotzé, G. Marsden, G. Lindgaard, J. Wesson, M. Winckler (Eds.), *Human-Computer Interaction – INTERACT 2013*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 246–263.
- [38] P. Dragicevic, *Fair Statistical Communication in HCI*, Springer International Publishing, Cham, 2016, pp. 291–330. doi : 10.1007/978-3-319-26633-6_13.
URL https://doi.org/10.1007/978-3-319-26633-6_13
- [39] M. R. Masliah, P. Milgram, Measuring the allocation of control in a 6 degree-of-freedom docking experiment, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '00*, Association for Computing Machinery, New York, NY, USA, 2000, pp. 25–32. doi : 10.1145/332040.332403.
URL <https://doi.org/10.1145/332040.332403>
- [40] W. S. Zwicker, Introduction to the theory of voting, in: F. Brandt, V. Conitzer, U. Endriss, J. Lang, A. D. Procaccia (Eds.), *Handbook of Computational Social Choice*, Cambridge University Press, 2016, Ch. 1.
- [41] A. H. Copeland, A reasonable social welfare function, Tech. rep., mimeo, 1951. University of Michigan (1951).
- [42] D. G. Saari, V. R. Merlin, The copeland method: I.: Relationships and the dictionary, *Economic Theory* 8 (1) (1996) 51–76.
URL <http://www.jstor.org/stable/25054952>
- [43] K. Kin, B. Hartmann, T. DeRose, M. Agrawala, Proton: Multitouch gestures as regular expressions, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12*, Association for Computing Machinery, New York, NY, USA, 2012, pp. 2885–2894. doi : 10.1145/2207676.2208694.
URL <https://doi.org/10.1145/2207676.2208694>

- [44] K. Kin, B. Hartmann, T. DeRose, M. Agrawala, Proton++: A customizable declarative multitouch framework, in: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology, UIST '12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 477–486. doi:10.1145/2380116.2380176. URL <https://doi.org/10.1145/2380116.2380176>
- [45] J. O. Wobbrock, A. D. Wilson, Y. Li, Gestures without libraries, toolkits or training: A \$1 recognizer for user interface prototypes, in: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, UIST '07, Association for Computing Machinery, New York, NY, USA, 2007, pp. 159–168. doi:10.1145/1294211.1294238. URL <https://doi.org/10.1145/1294211.1294238>
- [46] Y. Li, Protractor: A Fast and Accurate Gesture Recognizer, Association for Computing Machinery, New York, NY, USA, 2010, pp. 2169–2172. URL <https://doi.org/10.1145/1753326.1753654>
- [47] L. Anthony, J. O. Wobbrock, A lightweight multistroke recognizer for user interface prototypes, in: Proceedings of Graphics Interface 2010, GI '10, Canadian Information Processing Society, CAN, 2010, pp. 245–252.
- [48] R.-D. Vatavu, L. Anthony, J. O. Wobbrock, Gestures as point clouds: A \$p recognizer for user interface prototypes, in: Proceedings of the 14th ACM International Conference on Multimodal Interaction, ICMI '12, Association for Computing Machinery, New York, NY, USA, 2012, p. 273–280. doi:10.1145/2388676.2388732. URL <https://doi.org/10.1145/2388676.2388732>
- [49] R.-D. Vatavu, L. Anthony, J. O. Wobbrock, \$q: A super-quick, articulation-invariant stroke-gesture recognizer for low-resource devices, in: Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '18, Association for Computing Machinery,

New York, NY, USA, 2018. doi : 10.1145/3229434.3229465.

1065 URL <https://doi.org/10.1145/3229434.3229465>

- [50] C.-C. Lai, L.-W. Kuo, Size effects and scale effects on the usability of tablets in finger pointing and dragging tasks, in: F. Rebelo, M. Soares (Eds.), *Advances in Ergonomics in Design*, Springer International Publishing, Cham, 2016, pp. 109–120.