TouchTokens: Guiding Touch Patterns with Passive Tokens

Rafael Morales Gonzalez[2,1,4] **Caroline Appert**[1,2,4] **Gilles Bailly**[3,4] **Emmanuel Pietriga**[2,1,4]
[1]Univ Paris-Sud & CNRS ; [2]INRIA
Orsay, France
[3]Telecom-ParisTech & CNRS
Paris, France
[4]Univ Paris-Saclay
France
morales@lri.fr appert@lri.fr gilles.bailly@telecom-paristech.fr emmanuel.pietriga@inria.fr

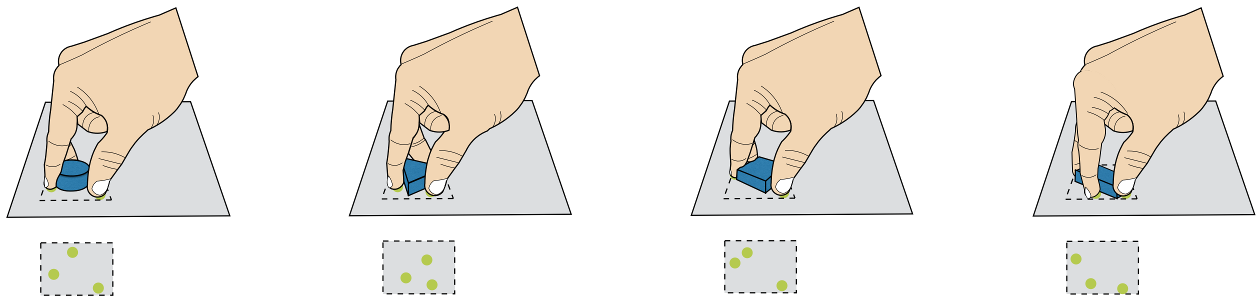

Figure 1. TouchTokens **are passive tokens that guide users' fingers to specific spatial configurations, resulting in distinguishable touch patterns.**

ABSTRACT

TouchTokens make it possible to easily build interfaces that combine tangible and gestural input using passive tokens and a regular multi-touch surface. The tokens constrain users' grasp, and thus, the relative spatial configuration of fingers on the surface, theoretically making it possible to design algorithms that can recognize the resulting touch patterns. We performed a formative user study to collect and analyze touch patterns with tokens of varying shape and size. The analysis of this pattern collection showed that individual users have a consistent grasp for each token, but that this grasp is user-dependent and that different grasp strategies can lead to confounding patterns. We thus designed a second set of tokens featuring notches that constrain users' grasp. Our recognition algorithm can classify the resulting patterns with a high level of accuracy (>95%) without any training, enabling application designers to associate rich touch input vocabularies with command triggers and parameter controls.

Author Keywords

Tangible interaction; Multi-Touch input

ACM Classification Keywords

H.5.2 : User Interfaces - Graphical user interfaces.

INTRODUCTION

The main characteristics of multi-touch gestures performed on the capacitive screens that typically equip tablets, smartphones, touchpads, as well as some tabletops, are the number of fingers involved and the individual trajectories of those fingers. Examples include 2- or 3-finger slide, and 2-finger pinch. But to the exception of a few research projects that consider touch points as chords [19, 21], interactive systems ignore the relative spatial configuration of contact points; what we call a *touch pattern*.

Our goal is to enable users to perform gestures based on a set of distinct touch patterns, thereby increasing the richness of input vocabularies for tactile surfaces. Our approach relies on physical guidance, as it would be unrealistic to expect touch patterns to be executed consistently across users, or even over time by the same user. As the literature suggests that users adopt grasp strategies that depend on the object to manipulate [39, 47], we investigate the potential of tangible tokens held on the surface to act as physical guides constraining the relative position of users' fingers.

We present TouchTokens, a novel interaction technique based on a set of easy-to-make passive tokens and a fast and simple recognition algorithm that can discriminate the unique touch pattern associated with each token in the set.[1] The approach features several advantages. First, physical tokens can provide space-multiplexed input by associating different controllers with different functions [18]. Second, tokens can alleviate issues related to discovery, exploration and learning inherent to gesture-based interaction [56]. Finally, tokens provide haptic feedback that promotes eyes-free interaction [28].

TouchTokens make it easy to implement applications that combine multi-touch and tangible input at low cost. Such a combination has the potential to foster collaboration, support distributed cognition, and enhance the user experience [1, 29, 46]. As opposed to other tangible systems that require electronic instrumentation (e.g., [9, 34]) or specific conductive material (e.g., [17, 33]), our system relies on an algorithm

Rafael Morales González, Caroline Appert, Gilles Bailly, and Emmanuel Pietriga. 2016. TouchTokens: Guiding Touch Patterns with Passive Tokens. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 4189-4202.

http://dx.doi.org/10.1145/2858036.2858041

[1]Implementations of the algorithm and vector descriptions of the tokens ready for 3D-printing or laser-cutting are available at https://www.lri.fr/~appert/touchtokens/.

that relies on standard multi-touch APIs and on passive tokens thats can be made of any non-conductive material, including wood or transparent acrylic.

We performed a formative user study to collect touch patterns, in which participants had to grasp and manipulate a set of twelve tokens of varying shape and size on a tabletop surface. The analysis of this pattern collection showed that people grasp the same token consistently across trials, but that it is quite difficult to identify a set of tokens and to design a robust recognition algorithm that works for all users. The two main sources of confusion are that different users may adopt different grasp strategies for the same token, and that one user may adopt the same strategy for distinct tokens. Based on these observations, we designed a second set of six tokens featuring notches that constrain users' grasp. These notches are designed to ensure a comfortable grasp while serving two purposes: 1) minimizing, for a given token, the variability of the contact points' relative position; and 2) maximizing the distinctiveness of touch patterns. We performed a summative study in which participants had to grasp and manipulate this set of tokens on both a tabletop and a tablet. Results show that our algorithm recognizes these touch patterns with an accuracy higher than 95%, and does so without any training or calibration. Application designers can map the gestures performed with these tokens to any command or parameter control, as illustrated in the examples introduced before the concluding discussion about limitations and future work.

RELATED WORK

TOUCHTOKENS makes use of physical tokens to augment the power of expression of multi-touch input, building upon tangible interaction and touch input research. Our review of related work is structured accordingly, giving an overview of projects that considered tangible tokens above interactive surfaces, or leveraged the power of expression of touch input.

Tangible tokens for tactile surfaces

Some tactile surfaces rely on *diffuse illumination*, which makes it possible to recognize both objects and hands in contact with the surface, using computer-vision algorithms to analyze the frames captured by IR cameras. Such techniques have been used, e.g., to track mice and keyboards [24] or to design physical widgets [50]. The Conté tool [49] is an artistic crayon that consists of an acrylic block that emits and reflects IR light. When tethered, its location and orientation can be tracked on diffuse illumination surfaces. Several projects rely on fiducial markers to ease the image-based analysis, such as the ReacTable [30, 31], which offers a tangible environment for music composition. Tokens can also be stacked on top of one another, using fiducial markers with transparent areas [3] or optical fiber bundles [5] to track them. Diffuse illumination hardware setups are somewhat bulky, however, and are thus mostly used for large surfaces such as tabletops.

Most touchscreens are *capacitive*: they detect a drop in capacitance when one or more fingers touch them. Various projects have investigated conductive objects. These objects contain a circuit of conductive material that links the areas that are in contact with the user's fingers to the areas that are in contact with the capacitive surface (the object's *"feet"*). As soon as the user touches an object, its feet become grounded and generate a drop in capacitance similar to a multi-touch pattern. Physical widgets [33] rely on this technique, as do physical button pads that can be clipped to the edges of a device [55], or more advanced objects that feature moving parts [17, 28] or that can be stacked [17]. Designing conductive tokens is challenging: the feet must be positioned carefully and the circuit must be stable so that the generated touch pattern can be recognized consistently. As capacitive screens have been designed for human fingers, properties such as the feet's minimal size and the minimal distance between two feet, which depend on the device, must be carefully chosen [55].

Other projects have explored more cost-effective ways of building conductive objects. Wiethoff et al. [51] use cardboard and conductive ink. This works well for low fidelity prototyping, but does not scale with real usages. Blagojevic et al. [11] report on a design experience where they have built a small set of geometric tools (ruler, protractor and set square) for a tabletop drawing application. They tested different construction strategies by combining different low-cost conductive materials (e.g., conductive ink, conductive foam, aluminium tape, copper wires). Their experience shows that making a physical tool conductive is quite difficult, as many factors have to be considered (consistent circuit, stability, friction with the screen, good grasp, etc.). In the end, the best design consisted of drilling holes in the tool and using conductive foam to cover the tool and fill the holes. In the panorama of capacitive tokens, PUCs [48] are an exception: they rely on the principle of mutual capacitance so as to be detected even when users do not touch them. However, most systems have to be augmented with an additional calibration clip to cheat the implemented adaptive filtering that tends to interfere with the PUCs' detection.

Some tangible systems work with *magnetic* tokens that modify the magnetic field incoming to the magnetometer built in mobile devices [8]. However, as a magnetometer reflects the sum of the magnetic fields it senses, supporting multiple tokens requires putting more than a simple magnet inside the tokens. Bianchi and Oakley [9] propose to use a more elaborate electronic system that features a motor to make the mounted magnet spin at a specific frequency. Putting a grid of Hall sensors behind the surface, Liang et al. [34] get a 2D image of the magnetic field that can be analyzed to track the location and orientation of objects above the surface. Each object can also be shielded with a case made of galvanized steel to avoid attraction and repellence effects between several tokens [35].

Multi-touch input and power of expression

Researchers have considered several avenues to increase the power of expression of touch input, including finger identification, finger pressure, or finger impact in order to multiplex input by assigning one command per finger, pressure level, or impacting zone. Finger identification relies on pattern recognition techniques coupled with advanced sensors such as fingerprint scanners [43] or fiber optic plates [27]. Projects such as SimPress [7] or FatThumb [12] capture the size of the finger's contact area and assign two different meanings to a soft tap and a hard tap. It is even possible to capture both the normal and tangential components of the force applied on the surface using extra pressure sensors [25]. Identification and amount of pressure of the finger in contact can also be assessed by classifying muscle activity in the forearm [6]. Finally, TapSense [22] discriminates which part of the finger (nail, knuckle, pad or tip) hits the surface by using acoustic sensing. With the exceptions of SimPress and FatThumb, that capture two pressure levels based on the size of the contact area, all of the above techniques rely on tactile surfaces that are augmented with additional sensors.

Some systems make use of whole-hand gestures (e.g., horizontal vs. vertical hand, straight vs. curved hand) for manipulating virtual objects [16, 40, 52, 53], or invoking virtual tools by mimicking the hold of their physical counterparts [4, 23]. Most of these projects rely on tactile surfaces that give access to the shape of the whole contact region, and cannot run on regular capacitive surfaces, which have been developed for finger input and consequently deliver standard point-based multi-touch coordinates only. One notable exception is the TouchTools system [23] that uses machine learning to recognize up to seven touch patterns associated with seven hand postures on a capacitive screen. A few systems can also recognize chord gestures. Finger-Count [2] counts the number of contact points on the surface. Arpège [21] supports more chords by relying on the contact points' relative position. The technique requires per-user calibration to record the fingers' natural position when the hand rests in a comfortable posture.

TOUCHTOKENS take a different approach and does not make the assumption that the fingers' relative position is always the same. The technique relies on *different relative finger positions* that users would adopt naturally when grasping a tangible token on a surface. Recognizing typical hand postures when people grasp objects has been investigated in experimental psychology to identify everyday objects and then infer users' activities (such as holding a mug or typing at the keyboard) [39]. Experimental studies show that it is possible to distinguish objects that differ in their size [14], shape (e.g., cylinder, pyramid, etc.) [42] or both [39, 47]. However, these studies assume that the system provides access to the whole hand posture, using advanced motion capture systems that can provide the position and orientation of all hand joints.

TOUCHTOKEN

The primary objective of TOUCHTOKENS is to guide the *registration pose* [19] of multi-touch gestures on an interactive surface. TOUCHTOKENS take advantage of users' ability to grab physical objects in the real word. Our idea is that the geometry (shape and size) of an object impacts how users grab it. Different objects will thus have different touch patterns on the tactile surface, which can be discriminated. Touch patterns are recognized at registration and remain active until all contact points have left the surface. In particular, users can *relax* their grasp in the *execution phase* of their gesture [54], thus reducing finger occlusion and enabling a larger range of motion. In this section, we describe how to build a system based on TOUCHTOKENS. Applications and limitations of the approach are discussed at the end of the paper.

Fabrication

TOUCHTOKENS require neither embedding electronics in the tokens nor augmenting the tactile surface with additional hardware (such as, e.g., a computer vision system), which makes setup easy. Tokens can be built from any non-conductive material such as wood, plastic, metal or glass, since the system only relies on the fingers' relative position, which is already provided by the tactile surface. This flexibility allows designers to easily prototype and test different TOUCHTOKENS variants with a 3D printer or a laser cutter. In particular, designers have a lot of control on the tokens' appearance. For tokens that have permanent roles associated with them, interface designers can engrave an icon or a label on them, or use a specific color. For temporary associations, end-users could adopt more volatile solutions, such as adding stickers or writing with an erasable pen if the chosen material affords it (e.g., pencil on a wooden token). Tokens can also be made of transparent material such as glass or acrylic, to avoid occluding the content displayed on the tactile surface.

Recognition

When grabbing a token with more than two fingers in contact with the surface, TOUCHTOKENS can infer its identity, and thus the corresponding registration pose, from the relative spatial configuration of the touch points. The recognition engine is initialized with one or more typical touch patterns per token and, when a touch pattern of at least three points occurs, the algorithm computes the distance between this input pattern and the set of template patterns. The recognized token is the one associated with the template that minimizes this distance metric.

Computing the distance between two touch patterns (input I:$\{I_1, ..., I_n\}$ and template T:$\{T_1, ..., T_n\}$) is not straightforward, however. First, most tactile surfaces do not provide finger identification. Second, tokens have an arbitrary orientation on the surface. Figure 2 illustrates how our algorithm processes touch patterns in order to identify the *best alignment* between the reference template and actual input patterns, from which the distance is computed.

The key steps for identifying the best alignment are as follows: (1) compute the centroid C_I of the three (or more) touch points; (2) generate all sequences of touchpoint labels (permutations) so that their IDs always appear in counterclockwise order; (3) rotate all these touch patterns so as to align vector $\overrightarrow{C_I I_1}$ with the x-axis. (4) The algorithm then translates touch patterns to align the input (C_I) and template (C_T) centroids. (5) It finally pairs the points in the permutation with the template's points in order to compute the distance, simply by summing all distances between paired points. The distance between reference template and actual input is given by the *best input alignment*, which is the permutation that minimizes this distance metric.

A typical implementation of the recognition engine amounts to about a hundred lines of code, and will work on any capacitive surface. The engine relies on simple geometrical features, which makes it easier to understand recognition errors compared to less transparent techniques such as those based

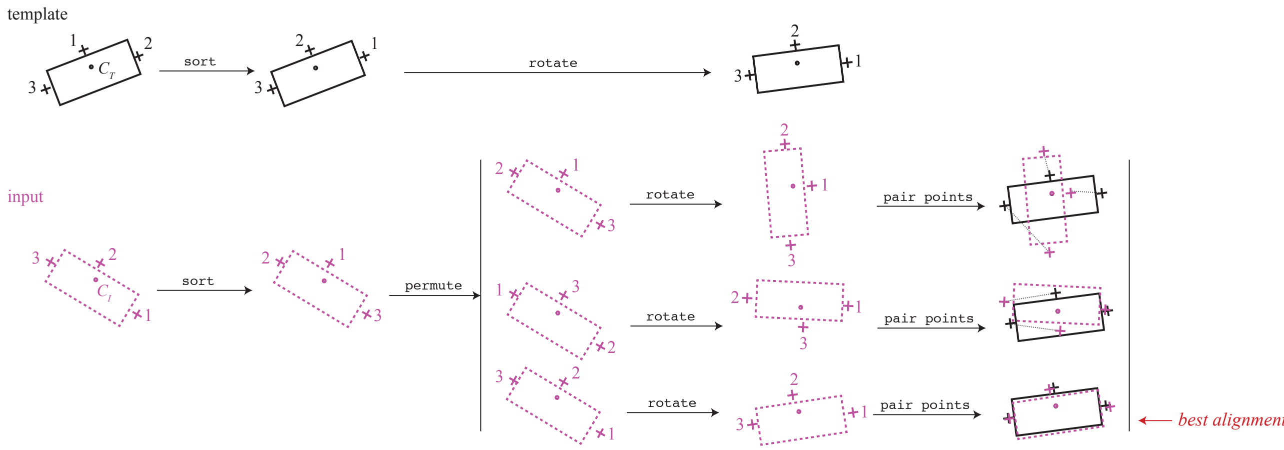

Figure 2. Template and input touch pattern alignment process.

on machine learning, that work as black boxes. The algorithm is very fast: recognition time scales linearly with the number of candidate templates. A Java implementation will be made available publicly, featuring both TUIO and Android APIs[2].

REGULAR TOKENS

TOUCHTOKENS rely on the hypothesis that the geometry of tokens impacts how users grasp them, resulting in distinguishable touch patterns. In order to test this hypothesis and identify a set of tokens that can actually be discriminated, we first ran a formative study in which participants had to grasp a set of twelve tokens that vary in shape and size.

Experiment design

Token Set

We selected a set of $4 \times 3 = 12$ tokens (Figure 3) that vary in their shape (square, circle, rectangle, and triangle) and size (3cm, 4cm and 5cm). The choice of size was informed by informal tests, taking into account both human and technological constraints. The tokens should remain comfortable to grasp with at least three fingers, which entails bio-mechanical constraints on the minimum and maximum token size. Capacitive surfaces also impose a minimal distance between finger tips, which will be seen as a single point if too close to one another. Our tokens are made of wood and are 6mm thick. We had initially considered tokens 3mm thick, but those were too difficult to grab. The tokens' corners are also slightly rounded so as to avoid sharp wedges that could have hurt participants.

Types of interaction

Participants are seated in front of the tabletop (at the center of the long edge) and perform a series of trials with the different tokens (Figure 4). As illustrated in Figure 5, the graphical display always features a progress bar in the top-left corner and a picture of the token to use in the current trial in the top-right corner. The action to be done with the token depends on the type of interaction (INTERACTION). The *Global* condition operationalizes the case where users invoke a global command with the token (e.g., launching an app); the *Local*

[2]It is part of the earlier-mentioned supplemental material made available to reviewers.

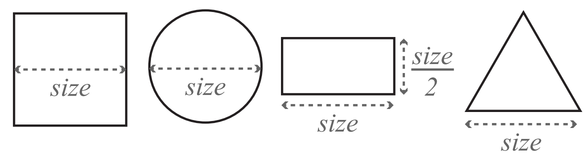

Figure 3. Set of tokens used in the first study (*size* $\in$ 3*cm*, 4*cm*, 5*cm*).

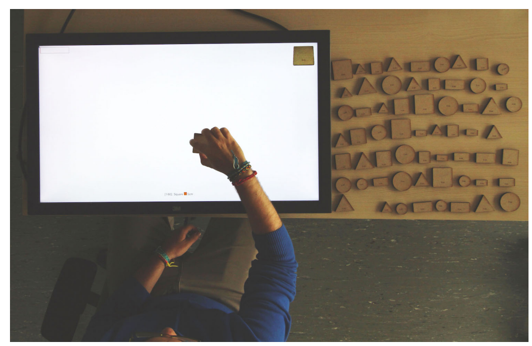

Figure 4. Experimental setup.

condition corresponds to the case where users apply a command at a specific location on screen (e.g., copying a graphical object); and the *Path* condition captures the case where users invoke a command and set its parameter value with a gesture (e.g., adjusting the opacity of a layer in a visualization). The progress bar indicates for how long participants have *dwelled*. It starts filling-in as soon as a stable touch pattern is detected on the surface. The dwell's duration depends on the type of interaction. If the number of fingers in contact changes, or if the touch pattern's centroid drifts away from its initial position by more than 30 pixels, the progress bar is reset and participants have to perform the trial again.

The experiment was divided into three phases, one per INTERACTION condition, always presented in the same order:

1. In the first phase (INTERACTION = *Global*), participants have to select the right token, put it anywhere on the tabletop, hold it with at least three fingers, and hold still for at least 1 second.

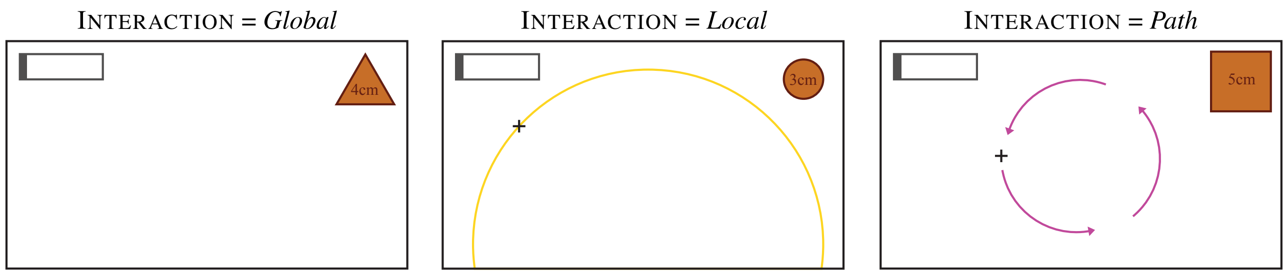

Figure 5. Types of interaction (INTERACTION)**.**

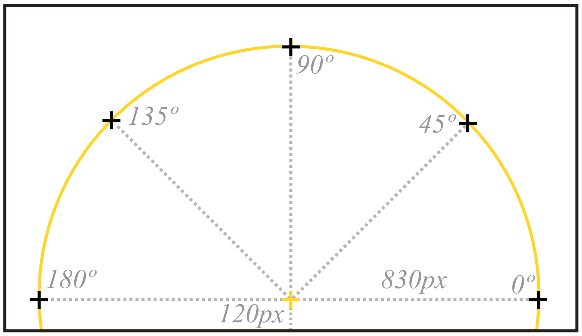

Figure 6. In the INTERACTION **=** ***Local*** **condition, participants have to put the token at a specific location** (LOCATION ∈ 0°, 45°, 90°, 135°, 180°)**.**

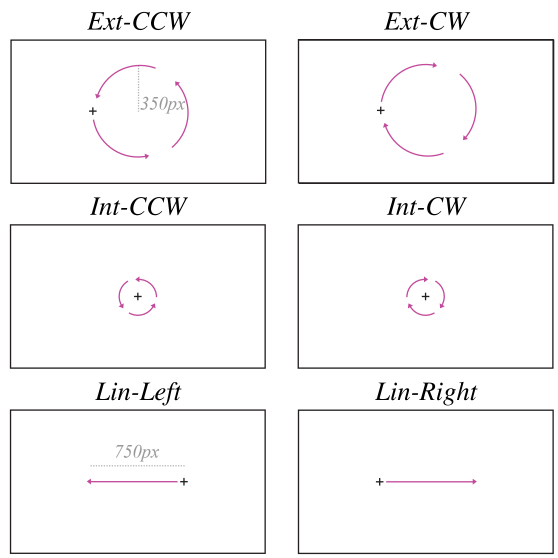

Figure 7. In the INTERACTION **=** ***Path*** **condition, the participant has to put the token on the surface and slide it along a specific path** (PATH ∈ **{*Ext-CCW*, *Ext-CW*, *Int-CCW*, *Int-CW*, *Lin-Left*, *Lin-Right*}).**

2. In the second phase (INTERACTION = *Local*), participants have to select the right token, put it on the cross (Figure 6), holding it still with at least three fingers for at least 1 second. The cross can be in five different LOCATION. These locations are chosen on a semi-circle roughly centered on the participant as in [38] (see Figure 6), as the token's location on the surface (relative to the participant) may influence the neutral hand posture and thus how the token is grasped. The distance between the touch pattern's centroid and the center of the cross must be at most 50px. If this distance is greater, the progress bar turns red and participants must perform the trial again.

3. In the third phase (INTERACTION = *Path*), participants must hold the token still with at least three fingers for a short period of 100ms. The background turns from gray to white. Participants then have to slide the token along the path indicated by purple arrows. In this condition, participants can plan a manipulation with the token, which may influence their initial grasp [38]. When sliding the token, they can lift some fingers but must keep at least one finger in contact with the surface. If they lift all fingers before having performed the whole gesture, the background turns back to gray and they have to start again. Figure 7 shows the six types of paths that participants had to follow with each token. We chose these tasks based on the taxonomy of multi-touch gestures from [37]. For external circular gestures (*Ext-CCW* and *Ext-CW*), participants have to slide the token along a clockwise or counterclockwise circular path. As soon as the touch pattern's centroid has completed one full circle, the background turns green and participants can proceed to the next trial. For internal circular gestures (*Int-CCW* and *Int-CW*), participants have to rotate the token around its center, as they would do with a physical circular knob. As soon as the touch pattern has been rotated by at least 45° around its centroid, the background turns green to indicate that the trial has been successfully completed. Finally, for linear gestures (*Lin-Left* and *Lin-Right*), participants simply have to slide the token to match the amplitude and direction indicated by the arrow.

Participants and Apparatus
Twelve volunteers (3 female), aged 23 to 33 year-old (average 26.5, median 25.5), participated in the experiment. The experiment software was running in full screen mode on a 3M C3266P6 capacitive screen (display dimensions: 698.4 x 392.85 mm, resolution: 1920 x 1080 pixels) placed horizontally on a desk (Figure 4). A digital video camera on a tripod recorded participants' hand and arm movements. The experimental software was developed in Java 2D (JDK 7) and ran on a Mac Pro 2.8 GHz Intel Quad Core with 16GB memory, running Mac OS X 10.7.5.

Procedure
Participants are seated at the center of the long side of the tabletop. They receive instructions detailing the goal of the experiment and the different experimental tasks they will

have to perform. In particular, the operator initially informs participants that the goal is to design a system that is able to recognize tokens based on users' grasp. He encourages them to be consistent in their grasp across trials with tokens that have the same shape. In order to identify which grasp is comfortable, the operator gives participants four tokens, one per shape with *size* = 4*cm* (*Square*$_4$, *Circle*$_4$, *Rectangle*$_4$ and *Triangle*$_4$), and asks them to manipulate each token a bit on the surface in order to choose a comfortable grasp. The operator then notes this grasp in his logs and the experiment starts.

As mentioned above, the experiment consists of three phases that are always presented in the same order:

- Phase 1 (INTERACTION = *Global*): 12 TOKEN × 5 repetitions = 60 trials. In this phase, the presentation order for the trials is randomized in order to observe if people are actually able to grasp the same token consistently across different trials that are not consecutive. To minimize the visual search time associated with identifying the right token to take, the operator printed 5 copies of each individual token and initially sorted the 60 tokens on the table, on the right side of the screen (Figure 4).
- Phase 2 (INTERACTION = *Local*): 12 TOKEN × 5 LOCATION × 2 repetitions = 120 trials. The order of TOKEN × LOCATION is randomized across participants. The 2 repetitions per TOKEN × LOCATION condition are presented one after another to limit the length of the experiment.
- Phase 3 (INTERACTION = *Path*): 12 TOKEN × 6 GESTURE × 2 repetitions = 144 trials. As in phase 2, the order of TOKEN × GESTURE conditions is randomized across participants, with the 2 repetitions presented one after another.

After completion of these three phases, participants receive a questionnaire where they have to give a comfort score for each of the twelve tokens. The questionnaire features 12 Likert-scale type questions where participants have to give a rating between 0 (not comfortable to grasp at all) to 5 (very comfortable). The overall procedure lasted about an hour.

Results

We first tested if participants' grasps of the different tokens can be distinguished using the recognition strategy described in the previous section. To that end, we train our recognition algorithm using the first three trials of Phase 1 as templates for each token. This training strategy corresponds to what a system relying on a light training phase would require.[3] We then evaluate our algorithm on the remaining trials, i.e., $(2 \times 12 + 2 \times 12 \times 5 + 2 \times 12 \times 6) \times 12$ participant = 3456 trials.

A χ^2 analysis reveals that both INTERACTION ($\chi^2(2, N = 3456) = 12, p = 0.002, \phi = 0.06$) and TOKEN ($\chi^2(11, N = 3456) = 109, p < 0.001, \phi = 0.18$) have a significant effect on RECOGNITION RATE. Figure 8 illustrates the observed differences between conditions. A finer analysis of TOKEN's effect on RECOGNITION RATE per INTERACTION shows that TOKEN

[3]We tested our algorithm with different training strategies to accommodate more variability (e.g., considering templates picked from the three experiment phases) but there was no clear gain compared against the training cost it would entail for end-users.

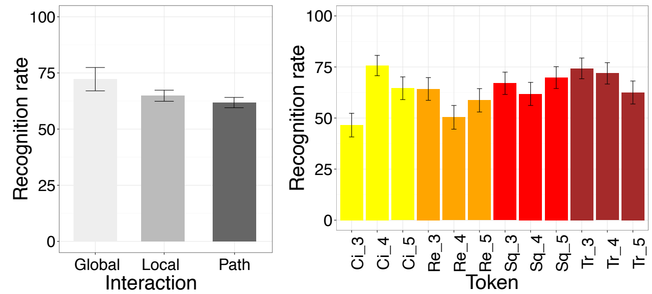

Figure 8. Recognition rate per INTERACTION (left) and per TOKEN (right). Error bars represent the 95% confidence interval.

has a significant effect on RECOGNITION RATE in all INTERACTION conditions (*Global*: $\chi^2(2, N = 288) = 25, p = 0.009, \phi = 0.3$, *Local*: $\chi^2(2, N = 1440) = 58, p < 0.001, \phi = 0.2$ and *Path*: $\chi^2(2, N = 1728) = 85, p < 0.001, \phi = 0.2$). The effect of secondary factors (LOCATION for *Local* and GESTURE for *Path*) on RECOGNITION RATE is not significant ($p = 0.4$ and $p = 0.2$).

We then wanted to investigate the impact of the token subset's size on recognition rate. In order to identify the largest number of grasps that can be accurately discriminated for each participant, we computed all possible subsets of tokens among the initial set of 12. The total number of subsets comprising at least two tokens (TOKENCOUNT >= 2) is:

$$\sum_{TokenCount=2}^{12} \binom{12}{TokenCount} = 2^{12} - 12 - 1 = 4083$$

For each subset, we ran our recognition algorithm with the same training strategy (only the first three trials from experiment phase INTERACTION = *Global*) in order to compute, for this subset, the recognition rate per participant. We observe that the per-subset recognition rate across participants exhibits a very high variability. For example, if we consider subsets that have 7 tokens (TOKENCOUNT = 7), the "worst" subset has a recognition rate of 63% on average across participants (worst-performing participant: 31%, best-performing participant: 98%), while the "best" subset has a recognition rate of 81% on average across participants (worst-performing participant: 57%, best-performing participant: 100%).

Recognition rate per participant

In order to test how many distinguishable grasps can be recognized per participant, we report the maximal recognition rate for each value of TOKENCOUNT $\in \{2, ..., 12\}$. If a participant P gets a maximal recognition rate R for TOKENCOUNT=N, this means that there exists at least one set of N tokens that are recognized with R% accuracy on average for participant P. Figure 9 reports these recognition rates for the best-performing and worst-performing participants, as well as the average over all participants. The charts illustrate that our algorithm can accurately discriminate a high number of grasps for some participants (the best-performing participant has a recognition accuracy higher than 90% for up to 10 tokens in all INTERACTION conditions), while it performs quite poorly for others (the worst-performing participant has

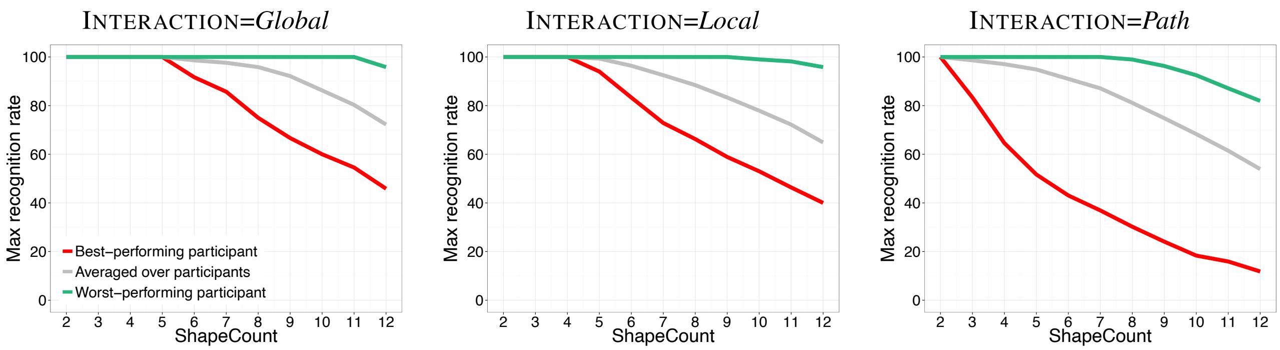

Figure 9. Recognition rate per INTERACTION × TOKEN

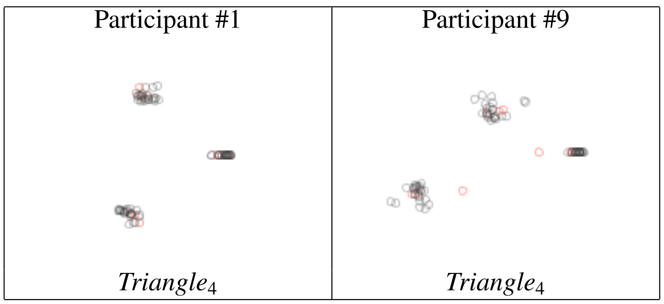

Figure 10. Touch patterns (aligned by our algorithm) for a participant who adopts very consistent grasps for token *Triangle*$_4$ (left) and for a participant who adopts varying grasps (right). Red dots belong to touch patterns that are used as templates.

a recognition accuracy lower than 90% even for sets of only three tokens in condition INTERACTION = *Path*). This variability comes from two sources: *intra-grasp variability* and *inter-grasp similarity*.

Figure 10 displays the 27 touch patterns we have collected for *Triangle*$_4$ for two participants. It illustrates two extreme levels of *intra-grasp variability*. Participant 1 (left) grasps token *Triangle*$_4$ in a very consistent manner, while Participant 9 (right) demonstrates much more variation in how he grasps it, challenging our recognition strategy. The second source of confusion comes from *inter-grasp similarity*: if a user chooses one grasp strategy for a given token that is very similar to the one he uses for another token in terms of similarity of the touch patterns, the two tokens will get confounded. Together, these two phenomena explain why we observe such a large variability across participants regarding the composition of the token sets that are recognized accurately.

Recognition rate between participants

Figure 9 reports the best set of tokens for each participant, and thus does not reflect the fact that the same subset of tokens can be very accurately recognized for one participant while it will be poorly recognized for another participant. We report the biggest sets of tokens that reach consensus among all our participants below (i.e., the sets of tokens that have a recognition accuracy of at least 90% for all participants):

- for INTERACTION=*Global*, we find 6 sets of 5 tokens ;
- for INTERACTION=*Local*, we find 13 sets of 3 tokens ;
- for INTERACTION=*Path*, we find 6 pairs of tokens ;
- for all INTERACTION conditions undifferentiated, we find 8 sets of 3 tokens with an average of at least 90% accuracy for all participants.

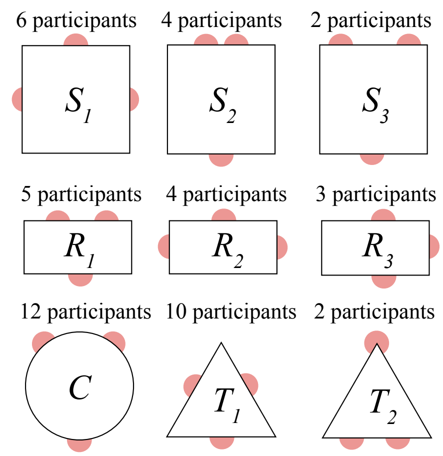

Figure 11. The nine grasp strategies observed in Experiment 1

Grasp strategies

Figure 11 summarizes the different grasp strategies that participants adopted for the different token shapes (extracted from an analysis of the operator's logs and video sequences recorded during the experiment). We observed that all participants use the same strategy for circles (C). Squares and rectangles receive less consensus, with three different strategies observed for each of them. The main strategy for squares uses three edges (S_1; 6/12). The two other strategies use only two edges, and differ in the distance between the two fingers on the same edge: small (S_2; 4/12) or large (S_3; 2/12). For rectangles, one strategy uses the two long edges only (R_1; 5/12). The two other strategies use three edges: two contact points on the short edges (R_2; 4/12) or on the long edges (R_3; 3/12). One of the grasp strategy for triangles makes use of a corner (T_2; 2/12), which was quite surprising. Two participants adopted it, but actually rated it as very uncomfortable.

To understand what kind of confusions occur in the recognition process, we implemented a visualization that displays all collected touch patterns using the best alignment computed by our recognition algorithm (Figure 10 was built with this tool). We computed the confusion matrix by considering the

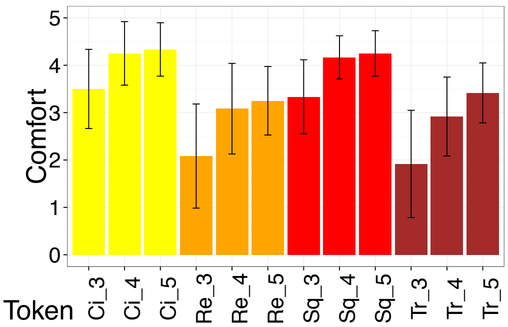

Figure 12. Comfort score per token. Error bars represent the 95% confidence interval.

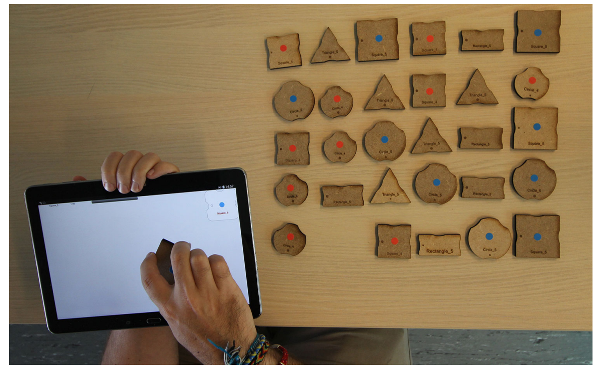

Figure 14. Experimental setup in the *Tablet* condition.

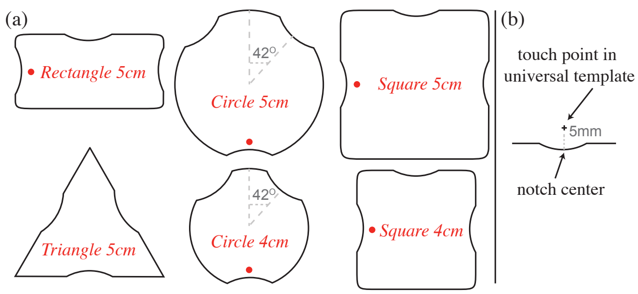

Figure 13. (a) The 6 tokens with notches. (b) The touch point's location in the template is offset by 5mm along the normal to the token's edge.

27 types of touch patterns (3 *size* × 9 grasp strategies). The visualization tool was a good complement to the confusion matrix as (1) some confusions do not appear in the matrix if a template for one token is too close to a template for another token; and (2) the different grasp strategies were not adopted the same number of times, leading to numbers in the confusion matrix that could not be compared in an absolute manner. From this analysis, we draw a few take-away messages. The flat isosceles triangle of grasp strategy R_2 is very representative and well-recognized. T_2 is also representative, but is too uncomfortable to be further considered. In contrast, some postures are difficult to distinguish. For instance, touch patterns R_1 and R_3 often form an equilateral triangle similar to the one of T_1. Finally, S_1 and C can also cause confusions.

TOKENS WITH NOTCHES

Our foundational hypothesis was that physical tokens constrain users' grasp in a consistent manner, which leads to consistent touch patterns that can be recognized with a high level of accuracy. The results of our formative experiment revealed that our hypothesis was only verified for some participants. We also observed significant variations in grasp strategies among users, which means that a set of tokens that works well for one user might not work so well for another user. As we aim at devising a solution that works effectively for all users in a consistent manner, we investigated a solution to decrease the different sources of variability.

We designed a new set of tokens, illustrated in Figure 13, similar to those considered in the formative study, but that feature notches. The purpose of these notches is to afford a particular grasp strategy, i.e., to suggest a specific way of positioning the fingers to grab a given token. The design of these tokens was guided by the following requirements. We wanted the token set to feature a wide range of shapes, as tokens should be easy to identify by visual and tactual perception [36]. Sets that feature different shapes also provide better mnemonic cues, making it easier for users to remember token-command associations. Finally, the tokens should remain comfortable to grasp. Based on these requirements, we picked the most comfortable size for each shape (5cm), and added the circular and square tokens of 4cm, which were also rated as very comfortable (Figure 12). We limited our summative study to this set of six tokens which, together with all token manipulation gestures, already provides a rich input vocabulary.

The grasp strategies observed during our formative experiment (Figure 11) informed the positioning of notches on token shapes. The notches' dimensions were refined through trial and error: narrow and deep notches introduce corners under finger tips, which make them uncomfortable; large and shallow notches are more comfortable, but introduce tangential variability in finger position. Our final design tries to strike a balance, and consists of notches 15mm wide and 1.5mm deep. Tokens whose shape afforded variable grasp strategies in the previous experiment feature a dot that indicates where to put the thumb, as illustrated in Figure 13-a.

These new tokens are designed to strongly constrain how users grasp them. We hypothesize that this will result in significantly reduced level of variability, which should enable our approach to work without any training. For each token, we compute a representative touch pattern that will act as a *universal template* for all users. The touch pattern is derived from the notches' position, slightly offset from the token's edge along the normal to that edge, so as to better capture users' grasp (Figure 13-b). The exact value of this offset (5mm) is calculated from the average offset measured in trials performed with circular tokens in the previous experiment, comparing the radius of the circle that passes through the three touch points with the radius of the actual physical token. (The precise, vector-based description of these tokens, ready for laser-cutting or 3D printing, will be distributed publicly and is also part of the earlier-mentioned supplemental material available to reviewers.)

EXPERIMENT

We ran a controlled experiment to test users' ability to manipulate tokens with notches, and to evaluate our algorithm's accuracy when provided with the above-mentioned universal templates in combination with this particular kind of tokens. The experimental design is similar to that of the previous study, but uses the set of 6 tokens of Figure 13. We also include an additional DEVICE condition: participants perform the tasks on both the tabletop and a tablet. Because of the smaller size of the tablet, we exclude the *Local* condition when DEVICE=*Tablet*, as the different locations (Figure 6) are clearly too close to one another to impact users' grasp. Contrary to our formative experiment, participants did not receive any other instructions than to grasp the tokens using the notches. In particular, the operator never asked them to adopt a consistent grasp across trials for a given token.

Experimental design

Procedure

Half of the participants started with the *Tabletop*, while the other half started with the *Tablet*. The strategy for counterbalancing the presentation order of trials is exactly the same as in the first experiment. The only difference lies in the *Tablet* condition, in which participants only performed *Global* and *Path* tasks (in this order), but not the *Local* task.

In the *Tabletop* condition, we collected 12 participants × 6 TOKEN × (5 [*Global*] + 2 × 5 LOCATION [*Local*] + 2 × 6 GESTURE [*Path*]) = 1944 trials. In the *Tabletop* condition, we collected 12 participants * 6 TOKEN × (5 [*Global*] + 2 × 6 GESTURE [*Path*]) = 1224 trials.

Participants & Apparatus

12 volunteers (3 female), aged 23 to 39 years-old (average 26.4, median 24.5), one left-handed, participated in this experiment. Five of them had participated in the previous study. The experimental setup for the *Tabletop* condition was exactly the same as in the previous experiment. In the *Tablet* condition, participants were seated at the same table, but had to hold the tablet during the whole experiment, as illustrated in Figure 14. The tablet (Samsung GT-P5110 Galaxy Tab 2) had a 256.7 x 175.3 mm display area with a resolution of 1280 x 800 pixels.

Results

As illustrated in Figure 15, the recognition rate in both DEVICE conditions is very high: 98.7% on the *Tabletop* and 99.3% on the *Tablet*. A χ^2 analysis reveals that the effect of INTERACTION on RECOGNITION RATE is significant neither in the *Tabletop* condition ($p = 0.8$) nor in the *Tablet* condition ($p = 0.3$). However, TOKEN has a significant effect in both DEVICE conditions (*Tabletop*: $\chi^2(5, N = 1944) = 30, p < 0.001, \phi = 0.12$ and *Tablet*: $\chi^2(5, N = 1224) = 30, p < 0.001, \phi = 0.16$)). Actually, in the *Tabletop* condition, the RECOGNITION RATE is a bit lower for *Circle*$_4$ (95.6%) than it is for all other tokens (> 98.7%). The same is true for token *Square*$_4$ (96.5%) in comparison with all other tokens (> 99%) in the *Tablet* condition.

Interestingly, even if we realized a posteriori that the thumb marker (dot) is meant for right-handed users, our left-handed participant did not have any trouble manipulating the tokens.

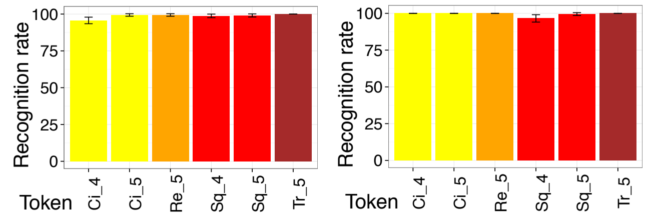

Figure 15. Recognition rate per TOKEN in the *Tabletop* (left) and *Tablet* (right) conditions. Error bars represent the 95% confidence interval.

He simply put his thumb in the notch opposite to the dot, ignoring the latter. Of course, he was able to do so because our tokens feature an axis of symmetry. However, we expect that TOUCHTOKENS's approach can be used for arbitrary-shape tokens, including some that would not feature a symmetric touch pattern. In that case, users can still flip them to accommodate their handedness, provided that the tokens are flat. If a token cannot be flipped easily, a solution would consist in designing two variants: same shape but pattern of notches mirrored. When the pattern cannot be mirrored because of the shape's geometry, it is still possible to design two patterns, one for each handedness.

APPLICATIONS

The above results show that, using tokens with notches, it is possible to build robust applications that will take advantage of both gesture-based and tangible interaction. Application domains that would benefit from such type of input are quite varied and have already been discussed in the literature, including: geographical information systems [32], database querying [29, 45], information management [41, 44] and music composition [30]. We developed a set of proof-of-concept applications[4] to illustrate the different roles that TOUCHTOKENS can play in an interactive system (see Figure 16).

TOUCHTOKENS can act as *controllers* or *filters*, and can be used to manipulate both the content of an application or the presentation of this content. For instance, they can be used to adjust the parameters of a visualization, enabling users to focus more on the result of their actions as the manipulation of physical tokens decreases the demand on visual attention [45]. We have developed a simple scatterplot visualization in D3 [13] to illustrate this idea. The different categories in the data (e.g., countries grouped by continent) are associated with different symbols (which have distinct shapes and colors), as is typically the case when visualizing multivariate datasets. One TOUCHTOKEN, with matching shape and color, is associated with each category and can be used to adjust the visual representation of the corresponding data points in the scatterplot: changing their size by rotating the token, and their opacity by sliding it. TOUCHTOKENS can be transparent, in which case they will typically be used as physical see-through tools [10, 15], altering the content that falls below the token (e.g., filtering) or changing its visual attributes (e.g., rendering). For instance, we have developed a simple mapping application in which tokens are associated with different layers. The tokens act as magic lenses [10] that

[4] All demonstrated in the companion video.

Figure 16. Proof-of-concept applications: access control, tangible magic lenses, character controllers in a game, data visualization.

reveal the corresponding layer while leaving the context untouched. See-through tools can also be used to move content in the workspace, as demonstrated in our simple game, where transparent tokens control the location and orientation of individual characters.

TOUCHTOKENS can also act as a receptacle for, or tangible representative of, digital content. Tokens then give access to the associated content [26]. One of our demo application illustrates how TOUCHTOKENS can be used for access control. Tokens can be used, e.g., to launch applications whose icons are otherwise invisible or disabled on the tablet's homescreen, enabling the device to be shared with family (parental control) and friends with some restrictions. Access to private content can be made even more secure by requiring that the token be put in a specific location, or that a particular gesture be performed with it. Our last application demonstrates the use of TOUCHTOKENS as digital containers. Users can reify photo albums into tokens, and add a picture to an album by holding the corresponding token above it. They can then display an album's content as a grid of thumbnails by rotating the token on the surface, or launch a slideshow by sliding it.

DISCUSSION AND FUTURE WORK

Main findings

We ran a formative experiment to investigate the possibility of recognizing individual tokens by categorizing their associated touch patterns. We were hypothesizing that differences in token shape and size might be sufficient to accurately discriminate those patterns. Our results revealed significant inter-user variability in terms of accuracy: our algorithm can recognize up to ten touch patterns with more than 90% accuracy for some users, while for other users, its accuracy falls down as soon as three or more tokens are in the set. This variability comes from two sources: 1) some users employ different grasp strategies for the same token; 2) some users employ grasp strategies for different tokens that yield very similar touch patterns.

Based on these observations, we then designed a set of six tokens featuring notches aimed at reducing this variability while remaining comfortable to grasp. A summative experiment showed that with this set of tokens, our recognizer has a minimum accuracy over all participants higher than 95% (avg. 98%), and this without any training. Augmented with notches, TOUCHTOKENS offer a low-cost, yet reliable, solution for enabling tangible interaction on multi-touch surfaces.

As mentioned earlier, we make this recognizer freely available, along with vector-based templates for the tokens.

Alternative recognition strategies

Our algorithm is fast, robust, and easy to implement. It also features the best recognition rate among all alternatives that we implemented and tested on the data collected during our formative study. Alternative approaches we considered led to significantly poorer performance. In particular, we tested k-Nearest-Neighbour (k=1 and k=3) and SVM algorithms, using both raw data and describing features. The raw data was pre-processed to make it independent from rotation angle and finger identification. The describing features we considered included the touch envelope's area, as well as various descriptive statistics (min, max, mean, median and standard deviation) for measures such as point-centroid distance, distance between successive points, distance between any pair of points, etc. These machine learning approaches yielded recognition rates ranging from 50% to 85% per participant. Compared to this, the analytical approach detailed in this paper, which consists in aligning touch patterns using their centroid as a reference point, works much better. We also considered using as a reference point the center of the best-fit circle (i.e. the circle that passes through three touch points while minimizing the distance to all remaining points) rather than the centroid, but results were slightly worse. The recognition rate was lowered by about 3% on average.

Future work

After this first investigation, we plan to study more systematically the limits of our approach, to see how it can scale to larger sets of tokens and/or to tokens that have varying geometries. We want to better characterize the minimal difference between two touch patterns, in order to be able to *automatically* position notches that meet our requirements: create tokens that are comfortable to grasp and that our algorithm can recognize with a high level of accuracy.

We also plan to conduct a fine-grained analysis of the fingers' traces on the surface at the precise moment they are lifted off. We want to investigate if these traces provide enough data to find out whether the token is still on the surface or not. Indeed, when lifting her fingers off the surface, the user might leave the token on it, or she might remove it. This would allow us to support additional interactions, such as when placing multiple tokens on the tactile surface to express, e.g., layout and alignment constraints [20] or advanced database queries with networks of tokens [29].

REFERENCES

1. Leonardo Angelini, Denis Lalanne, Elise van den Hoven, Ali Mazalek, Omar Abou Khaled, and Elena Mugellini. 2015. Tangible Meets Gestural: Comparing and Blending Post-WIMP Interaction Paradigms. In *Proceedings of the 9th International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 473–476. DOI: http://dx.doi.org/10.1145/2677199.2683583

2. Gilles Bailly, Jörg Müller, and Eric Lecolinet. 2012. Design and Evaluation of Finger-count Interaction: Combining Multitouch Gestures and Menus. *Int. J. Hum.-Comput. Stud.* 70, 10 (Oct. 2012), 673–689. DOI: http://dx.doi.org/10.1016/j.ijhcs.2012.05.006

3. Tom Bartindale and Chris Harrison. 2009. Stacks on the Surface: Resolving Physical Order Using Fiducial Markers with Structured Transparency. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, 57–60. DOI: http://dx.doi.org/10.1145/1731903.1731916

4. Tom Bartindale, Chris Harrison, Patrick Olivier, and Scott E. Hudson. 2011. SurfaceMouse: Supplementing Multi-touch Interaction with a Virtual Mouse. In *Proceedings of the 5th International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, 293–296. DOI: http://dx.doi.org/10.1145/1935701.1935767

5. Patrick Baudisch, Torsten Becker, and Frederik Rudeck. 2010. Lumino: Tangible Blocks for Tabletop Computers Based on Glass Fiber Bundles. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, 1165–1174. DOI: http://dx.doi.org/10.1145/1753326.1753500

6. Hrvoje Benko, T. Scott Saponas, Dan Morris, and Desney Tan. 2009. Enhancing Input on and Above the Interactive Surface with Muscle Sensing. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, 93–100. DOI: http://dx.doi.org/10.1145/1731903.1731924

7. Hrvoje Benko, Andrew D Wilson, and Patrick Baudisch. 2006. Precise Selection Techniques for Multi-Touch Screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems (CHI '06)*. 1263–1272. DOI: http://dx.doi.org/10.1145/1124772.1124963

8. Andrea Bianchi and Ian Oakley. 2013. Designing Tangible Magnetic Appcessories. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction (TEI '13)*. ACM, 255–258. DOI: http://dx.doi.org/10.1145/2460625.2460667

9. Andrea Bianchi and Ian Oakley. 2015. MagnID: Tracking Multiple Magnetic Tokens. In *Proceedings of the 9th International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 61–68. DOI: http://dx.doi.org/10.1145/2677199.2680582

10. Eric A. Bier, Maureen C. Stone, Ken Pier, William Buxton, and Tony D. DeRose. 1993. Toolglass and Magic Lenses: The See-through Interface. In *Proceedings of the 20th Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '93)*. ACM, 73–80. DOI: http://dx.doi.org/10.1145/166117.166126

11. Rachel Blagojevic and Beryl Plimmer. 2013. CapTUI: Geometric Drawing with Tangibles on a Capacitive Multi-touch Display. In *Proceedings of INTERACT '13*. Springer, 511–528.

12. Sebastian Boring, David Ledo, Xiang 'Anthony' Chen, Nicolai Marquardt, Anthony Tang, and Saul Greenberg. 2012. The fat thumb: using the thumb's contact size for single-handed mobile interaction. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services companion (MobileHCI '12)*. ACM, 207–208. DOI: http://dx.doi.org/10.1145/2371664.2371711

13. Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. 2011. D3 Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics* 17, 12 (Dec. 2011), 2301–2309. DOI: http://dx.doi.org/10.1109/TVCG.2011.185

14. Bryan Buchholz, Thomas J Armstrong, and Steven A Goldstein. 1992. Anthropometric data for describing the kinematics of the human hand. *Ergonomics* 35, 3 (1992), 261–273.

15. Wolfgang Büschel, Ulrike Kister, Mathias Frisch, and Raimund Dachselt. 2014. T4 - Transparent and Translucent Tangibles on Tabletops. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '14)*. ACM, 81–88. DOI: http://dx.doi.org/10.1145/2598153.2598179

16. Xiang Cao, A.D. Wilson, R. Balakrishnan, K. Hinckley, and S.E. Hudson. 2008. ShapeTouch: Leveraging contact shape on interactive surfaces. In *Proceedings of the 3rd International Workshop on Horizontal Interactive Human Computer Systems (TABLETOP '08)*. IEEE, 129–136. DOI: http://dx.doi.org/10.1109/TABLETOP.2008.4660195

17. Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. 2012. CapStones and ZebraWidgets: Sensing Stacks of Building Blocks, Dials and Sliders on Capacitive Touch Screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2189–2192. DOI: http://dx.doi.org/10.1145/2207676.2208371

18. George W. Fitzmaurice, Hiroshi Ishii, and William A. S. Buxton. 1995. Bricks: Laying the Foundations for Graspable User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '95)*. ACM Press/Addison-Wesley Publishing Co., 442–449. DOI: http://dx.doi.org/10.1145/223904.223964

19. Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris, and Daniel Wigdor. 2009. 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)*. ACM, 165–172. DOI: http://dx.doi.org/10.1145/1731903.1731935

20. Mathias Frisch, Sebastian Kleinau, Ricardo Langner, and Raimund Dachselt. 2011. Grids & Guides: Multi-touch Layout and Alignment Tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 1615–1618. DOI: http://dx.doi.org/10.1145/1978942.1979177

21. Emilien Ghomi, Stéphane Huot, Olivier Bau, Michel Beaudouin-Lafon, and Wendy E. Mackay. 2013. Arpège: Learning Multitouch Chord Gestures Vocabularies. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '13)*. ACM, 209–218. DOI: http://dx.doi.org/10.1145/2512349.2512795

22. Chris Harrison, Julia Schwarz, and Scott E. Hudson. 2011. TapSense: Enhancing Finger Interaction on Touch Surfaces. In *Proceedings of the 24th ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 627–636. DOI: http://dx.doi.org/10.1145/2047196.2047279

23. Chris Harrison, Robert Xiao, Julia Schwarz, and Scott E. Hudson. 2014. TouchTools: Leveraging Familiarity and Skill with Physical Tools to Augment Touch Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 2913–2916. DOI: http://dx.doi.org/10.1145/2556288.2557012

24. Björn Hartmann, Meredith Ringel Morris, Hrvoje Benko, and Andrew D. Wilson. 2009. Augmenting Interactive Tables with Mice & Keyboards. In *Proceedings of the 22nd ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, 149–152. DOI: http://dx.doi.org/10.1145/1622176.1622204

25. Seongkook Heo and Geehyuk Lee. 2011. Force Gestures: Augmenting Touch Screen Gestures with Normal and Tangential Forces. In *Proceedings of the 24th ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 621–626. DOI: http://dx.doi.org/10.1145/2047196.2047278

26. Lars Erik Holmquist, Johan Redström, and Peter Ljungstrand. 1999. Token-Based Access to Digital Information. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing (HUC '99)*. Springer-Verlag, 234–245. http://dl.acm.org/citation.cfm?id=647985.743869

27. Christian Holz and Patrick Baudisch. 2013. Fiberio: A Touchscreen That Senses Fingerprints. In *Proceedings of the 26th ACM Symposium on User Interface Software and Technology (UIST '13)*. ACM, 41–50. DOI: http://dx.doi.org/10.1145/2501988.2502021

28. Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. 2012. Tangible Remote Controllers for Wall-size Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2865–2874. DOI: http://dx.doi.org/10.1145/2207676.2208691

29. Hans-Christian Jetter, Jens Gerken, Michael Zöllner, Harald Reiterer, and Natasa Milic-Frayling. 2011. Materializing the Query with Facet-streams: A Hybrid Surface for Collaborative Search on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 3013–3022. DOI: http://dx.doi.org/10.1145/1978942.1979390

30. Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. 2007. The reacTable: Exploring the Synergy Between Live Music Performance and Tabletop Tangible Interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. ACM, 139–146. DOI: http://dx.doi.org/10.1145/1226969.1226998

31. Martin Kaltenbrunner and Ross Bencina. 2007. reacTIVision: A Computer-vision Framework for Table-based Tangible Interaction. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. ACM, 69–74. DOI: http://dx.doi.org/10.1145/1226969.1226983

32. Hideki Koike, Wataru Nishikawa, and Kentaro Fukuchi. 2009. Transparent 2-D Markers on an LCD Tabletop System. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, 163–172. DOI: http://dx.doi.org/10.1145/1518701.1518728

33. Sven Kratz, Tilo Westermann, Michael Rohs, and Georg Essl. 2011. CapWidgets: Tangible Widgets Versus Multi-touch Controls on Mobile Devices. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, 1351–1356. DOI: http://dx.doi.org/10.1145/1979742.1979773

34. Rong-Hao Liang, Kai-Yin Cheng, Liwei Chan, Chuan-Xhyuan Peng, Mike Y. Chen, Rung-Huei Liang, De-Nian Yang, and Bing-Yu Chen. 2013. GaussBits: Magnetic Tangible Bits for Portable and Occlusion-free Near-surface Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 1391–1400. DOI: http://dx.doi.org/10.1145/2470654.2466185

35. Rong-Hao Liang, Han-Chih Kuo, Liwei Chan, De-Nian Yang, and Bing-Yu Chen. 2014. GaussStones: Shielded Magnetic Tangibles for Multi-token Interactions on Portable Displays. In *Proceedings of the 27th ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 365–372. DOI: http://dx.doi.org/10.1145/2642918.2647384

36. Jack M Loomis and Susan J Lederman. 1986. Tactual perception. In *Handbook of Perception and Human Performance*, K. R. Boff, L. Kaufman, and J. P. Thomas (Eds.). John Wiley & Sons.

37. Halla Olafsdottir and Caroline Appert. 2014. Multi-touch Gestures for Discrete and Continuous Control. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '14)*. ACM, 177–184. DOI: http://dx.doi.org/10.1145/2598153.2598169

38. Halla Olafsdottir, Theophanis Tsandilas, and Caroline Appert. 2014. Prospective Motor Control on Tabletops: Planning Grasp for Multitouch Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 2893–2902. DOI: http://dx.doi.org/10.1145/2556288.2557029

39. Brandon Paulson, Danielle Cummings, and Tracy Hammond. 2011. Object Interaction Detection Using Hand Posture Cues in an Office Setting. *Int. J. Hum.-Comput. Stud.* 69, 1-2 (Jan. 2011), 19–29. DOI: http://dx.doi.org/10.1016/j.ijhcs.2010.09.003

40. Jun Rekimoto. 2002. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. ACM, 113–120. DOI: http://dx.doi.org/10.1145/503376.503397

41. Jun Rekimoto, Brygg Ullmer, and Haruo Oba. 2001. DataTiles: A Modular Platform for Mixed Physical and Graphical Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. ACM, 269–276. DOI: http://dx.doi.org/10.1145/365024.365115

42. Marco Santello and John F Soechting. 1998. Gradual molding of the hand to object contours. *Journal of neurophysiology* 79, 3 (1998), 1307–1320.

43. Atsushi Sugiura and Yoshiyuki Koseki. 1998. A User Interface Using Fingerprint Recognition: Holding Commands and Data Objects on Fingers. In *Proceedings of the 11th ACM Symposium on User Interface Software and Technology (UIST '98)*. ACM, 71–79. DOI: http://dx.doi.org/10.1145/288392.288575

44. Brygg Ullmer, Hiroshi Ishii, and Dylan Glas. 1998. mediaBlocks: Physical Containers, Transports, and Controls for Online Media. In *Proceedings of the 25th Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. ACM, 379–386. DOI: http://dx.doi.org/10.1145/280814.280940

45. Brygg Ullmer, Hiroshi Ishii, and Robert J. K. Jacob. 2003. Tangible Query Interfaces: Physically Constrained Tokens for Manipulating Database Queries. In *Proceedings of INTERACT '03*. 279–286.

46. Consuelo Valdes, Diana Eastman, Casey Grote, Shantanu Thatte, Orit Shaer, Ali Mazalek, Brygg Ullmer, and Miriam K. Konkel. 2014. Exploring the Design Space of Gestural Interaction with Active Tokens Through User-defined Gestures. In *Proceedings of the SIGCHI conference on Human Factors in computing systems (CHI '14)*. ACM, 4107–4116. DOI: http://dx.doi.org/10.1145/2556288.2557373

47. Radu-Daniel Vatavu and Ionu Alexandru Zaii. 2013. Automatic Recognition of Object Size and Shape via User-dependent Measurements of the Grasping Hand. *Int. J. Hum.-Comput. Stud.* 71, 5 (May 2013), 590–607. DOI: http://dx.doi.org/10.1016/j.ijhcs.2013.01.002

48. Simon Voelker, Kosuke Nakajima, Christian Thoresen, Yuichi Itoh, Kjell Ivar Øvergård, and Jan Borchers. 2013. PUCs: Detecting Transparent, Passive Untouched Capacitive Widgets on Unmodified Multi-touch Displays. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '13)*. ACM, 101–104. DOI: http://dx.doi.org/10.1145/2512349.2512791

49. Daniel Vogel and Géry Casiez. 2011. Conté: Multimodal Input Inspired by an Artist's Crayon. In *Proceedings of the 24th ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 357–366. DOI: http://dx.doi.org/10.1145/2047196.2047242

50. Malte Weiss, Julie Wagner, Yvonne Jansen, Roger Jennings, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. 2009. SLAP Widgets: Bridging the Gap Between Virtual and Physical Controls on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, 481–490. DOI: http://dx.doi.org/10.1145/1518701.1518779

51. Alexander Wiethoff, Hanna Schneider, Michael Rohs, Andreas Butz, and Saul Greenberg. 2012. Sketch-a-TUI: Low Cost Prototyping of Tangible Interactions Using Cardboard and Conductive Ink. In *Proceedings of the 6th International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*. ACM, 309–312. DOI:http://dx.doi.org/10.1145/2148131.2148196

52. Daniel Wigdor, Hrvoje Benko, John Pella, Jarrod Lombardo, and Sarah Williams. 2011. Rock & Rails: Extending Multi-touch Interactions with Shape Gestures to Enable Precise Spatial Manipulations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 1581–1590. DOI: http://dx.doi.org/10.1145/1978942.1979173

53. Mike Wu and Ravin Balakrishnan. 2003. Multi-finger and Whole Hand Gestural Interaction Techniques for Multi-user Tabletop Displays. In *Proceedings of the 16th ACM Symposium on User Interface Software and Technology (UIST '03)*. ACM, 193–202. DOI: http://dx.doi.org/10.1145/964696.964718

54. Mike Wu, Chia Shen, Kathy Ryall, Clifton Forlines, and Ravin Balakrishnan. 2006. Gesture Registration, Relaxation, and Reuse for Multi-Point Direct-Touch Surfaces. In *Proceedings of the 1st International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06)*. IEEE, 185–192. DOI: http://dx.doi.org/10.1109/TABLETOP.2006.19

55. Neng-Hao Yu, Sung-Sheng Tsai, I-Chun Hsiao, Dian-Je Tsai, Meng-Han Lee, Mike Y. Chen, and Yi-Ping Hung. 2011. Clip-on Gadgets: Expanding Multi-touch Interaction Area with Unpowered Tactile Controls. In *Proceedings of the 24th ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 367–372. DOI: http://dx.doi.org/10.1145/2047196.2047243

56. Shumin Zhai, Per Ola Kristensson, Caroline Appert, Tue Haste Andersen, and Xiang Cao. 2012. Foundational Issues in Touch-Surface Stroke Gesture Design: An Integrative Review. *Found. Trends Hum.-Comput. Interact.* 5, 2 (Feb. 2012), 97–205. DOI: http://dx.doi.org/10.1561/1100000012