Conté: Multimodal Input Inspired by an Artist's Crayon

Daniel Vogel[1,2] and Géry Casiez[1]

[1]LIFL & INRIA Lille
University of Lille, FRANCE
gery.casiez@lifl.fr

[2]Cheriton School of Computer Science
University of Waterloo, CANADA
dvogel@uwaterloo.ca

ABSTRACT

Conté is a small input device inspired by the way artists manipulate a real Conté crayon. By changing which corner, edge, end, or side is contacting the display, the operator can switch interaction modes using a single hand. Conté's rectangular prism shape enables both precise pen-like input and tangible handle interaction. Conté also has a natural compatibility with multi-touch input: it can be tucked in the palm to interleave same-hand touch input, or used to expand the vocabulary of bimanual touch. Inspired by informal interviews with artists, we catalogue Conté's characteristics, and use these to outline a design space. We describe a prototype device using common materials and simple electronics. With this device, we demonstrate interaction techniques in a test-bed drawing application. Finally, we discuss alternate hardware designs and future human factors research to study this new class of input.

ACM Classification: H.5.2 Information Interfaces and Presentation: Input; Interaction styles

General terms: Human Factors

Keywords: pen, touch, gestures, tabletop, multimodal

INTRODUCTION

Touch interaction is arguably more immediate and natural in many situations, but fingers are imprecise [27] and difficult to write with [3,15]. Alternatively, using a pen (or stylus) makes writing more natural and pointing more precise [29]. Luckily, this does not need to be a unilateral choice; pen and touch can be used simultaneously [3,15,29,30]. However, without non-dominant hand coordination or graphical buttons, the pen *itself* supports few modes. This makes single-handed mobile usage difficult and reduces the number of combined touch and pen modes. When frequently switching between pen-oriented modes, such as drawing, handwriting, gestures, and lasso selection [29], this can hurt performance [18]. Inferring modes is difficult, and most users prefer explicit control [7]. Schemes for squeezing multiple *explicit* modes from a pen include adding barrel buttons and classifying pressure [22], tilt [26], barrel rotation [3], or grip [25]. But these can be error-prone and ambiguous. A simple way to add a second mode is by adding an "eraser," a second *contact* point. The pencil analogy lends intuition and users have explicit control.

Our work extends this idea of switching modes using contact points by referencing how artists use a *real* Conté crayon, a square prism-shaped drawing stick. Its faceted shape creates a variety of marks according to how it contacts paper: corner, edge, end, or side (Figure 1). A *digital* Conté crayon could even sense *which* corner, *which* edge, or *which* side. Thus, Conté builds on ToolStone [23], but with notable differences: ToolStone is a larger block shape, uses only side contacts, and is designed for non-dominant hand usage on an indirect tablet. Conté can function single-handed, yet span multiple precise pen-like input modes and modes similar to tangible handles [16]. For example: one corner for drawing and another for handwriting recognition; one end edge for stroke-based gestures and another for lasso selection; one side to reveal a tool palette and another to create guidelines. Artists blend real Conté drawings with their fingers, so combining digital Conté with multi-touch input is natural. By tucking Conté in the palm, same-hand touch may be interleaved and, similar to Manual Deskterity [15], a Conté-enabled mode may be further manipulated with bimanual touch. For example: rotating a guideline with Conté while selecting angle snap with touch.

Inspired by informal interviews with artists, we catalogue digital Conté's characteristics, and use these to outline a design space. A principal challenge is the creation of a functional device. We describe the hardware and software for our digital Conté stick which is compatible with diffuse infrared illumination tabletops. Our solution uses common materials, basic tools, and simple electronics, making it possible for others to build their own Conté. As future work, we discuss alternate device implementations.

With our prototype Conté device, we implemented several compatible interaction techniques in a test-bed drawing application. These specific examples taken from the larger design space demonstrate the feasibility and highlight practical aspects of Conté techniques. A user evaluation would be premature for this nascent device [13], but we close with a discussion of avenues for formal evaluation with an emphasis on fundamental human factors questions raised by this new class of input.

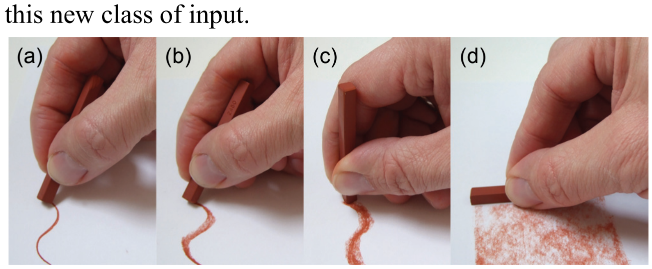

Figure 1. Drawing marks left by an artist's Conté crayon when using: (a) corner; (b) edge; (c) end; and (d) side.

RELATED WORK

Elegantly switching between multiple modes has been a challenge since the first pen input systems. The reader may recall the classic Sketchpad demonstration [31] where the operator is constantly "tickling" toggle switches. Later, DENIM [19] and Tivoli [20] state the problem clearly, and experiment with various techniques. In a study of "ink vs. gesturing" mode selection, Li et al. [18] found that pressing a hardware button with the non-dominant hand was fastest. However, each mode needs a button, and more than two modes may be necessary: Wu et al. [29] suggest 3 pen modes ("...write, annotate and sketch"); Brandl et al. add precise selection [4]; and Song et al. introduce 4 more [25].

Bimanual Touch and Pen Mode Selection. By combining a *single-touch* screen with a pen sensor Yee [30] demonstrates how non-dominant button touches, similar to Li et al.'s hardware button, can select among different pen tools. Wu et al. [29] and Brandl et al. [4] refine this further, by combining a pen with a *multi-touch* surface. Now modes may be selected according to the type of non-dominant hand contact. For example, touching two fingers changes from pen selection to pen inking [29], or a flat palm contact changes from free-form to poly-line pen drawing [4]. These combined touch and pen systems enable a broad continuum of interaction. On one extreme, pen and touch can act independently, like Yee's [30] and Brandl at al.'s [4] use of touch to pan the drawing surface while the pen writes. At the other end, pen and touch can coordinate very closely, such as Yee's [30] combined pen and touch zoom.

Hinckley et al.'s Manual Deskterity system [15] refines bimanual pen and touch coordination by considering the contextual relationship with direct manipulation objects (e.g. notes in a sketchbook application). When used alone, the pen always draws and the hand always manipulates, but when used in the context of an object, an implicit mode is entered. For example, depending on how an object is held with non-dominant touches and the object-relative motion of the pen, a pen stroke may cut like a knife, draw a straight line along an edge, or create a copy of the object. This greatly increases the number of modes in an elegant way, but various hand postures need to be recalled and performed unambiguously [2], and not all modes seem to have obvious bimanual contextual mappings. Consider how Hinckley et al. resort to a toolbar button to change the pen to a highlighter. Finally, along with Yee, Brandl et al., and Wu et al., two hands are required. Hinckley et al. do include redundant pen-only command access via radial menus, but acknowledge this adds syntactical complexity.

Unimanual Pen Mode Selection. Commercial pen input systems, such as the Wacom 6D Art Pen, provide multiple input dimensions which can be used for one-handed mode switching. Among one-handed techniques, Li et al. [18] find pressure and barrel button fastest, but with higher error rates (only beating conventional press-and-hold which was also slowest). Flipping an "eraser"-equipped pen is also slow, but with the lowest one-handed error rate. Li et al. only test binary mode selection, but by classifying continuous parameters like pressure [22], tilt [26], barrel roll [3], or even barrel grip [25], up to 6 modes can be supported. However, these can be ambiguous and error-prone due to interference from normal pen movements.

Mode Selection by Contact. Expanding on Fitzmaurice et al.'s flipbrick idea [8], Rekimoto and Sciammarella's ToolStone [23] is a block which is flipped and rotated to select between as many as 48 different modes. Although designed for non-dominant hand input on an indirect tablet, it illustrates how sensing the device-to-surface *contact* can be harnessed for mode selection. ToolStone focuses on face contacts; edge sensing is mentioned, but not explored. At $25 \times 40 \times 50$ mm, ToolStone is much larger than a pen but, in an elicitation study, Frisch et al. [10] observe some users using the side of a pen in a similar manner.

Summary. Conté builds on the ToolStone concept, but with notable differences. Conté has a much smaller form factor and is capable of detecting a more diverse set of contacts such as corners, short and long edges, sides, and ends. Although not a pen per se, when a corner or end edge contact is used, Conté approaches pen-like manipulation and precision. Thus, with an ability to switch modes based on contact, it directly addresses the unimanual pen mode selection problem. Compared to flipping an eraser-equipped pen, Conté has contact point adjacencies — clusters of different contact points at one end — with much shorter transition times compared to flipping a pen.

Also, unlike ToolStone, Conté is designed to work with multi-touch on a direct input display, a natural combination given how artists blend real Conté strokes with their finger. Dominant single-hand touch input can be interleaved by tucking Conté in the palm or setting Conté down. Bimanual pen and touch like Yee [30], Wu et al. [29], Brandl et al. [3], and Manual Deskterity [15] can also be achieved due to Conté's pen-like form factor. But, with Conté's multi-modal capability, the design space is expanded.

EXPLORATORY INTERVIEWS

We conducted interviews with two practicing artists to gain a better understanding of real Conté and its potential digital counterpart. The specific questions and tasks were:

1. *How are real Conté sticks used*? Participants create simple Conté drawings on paper;
2. *How might digital Conté perform fundamental tasks?* Participants write, draw, tap, and trace paths;
3. *How might digital Conté perform basic manipulations?* Participants change contact points, set the stick down and pick it up, and "tuck" it in their palm.

For question 1, participants used a real Conté stick. For questions 2 - 4, they used solid acrylic mock-ups of different sticks with display-like feedback enabled by a "Magic Drawing Slate" novelty toy (Figure 2b). This is a grey sheet of film loosely laid over a black waxy under-layer. Pressing against the film creates a blackish mark and the marks are erased by lifting the film. We found this basic feedback essential to feel how digital Conté may work for mark-making. We tested three different Conté mock-ups (Figure 2a): 6 × 6 × 64 mm (same as artist's Conté crayons), 8 × 8 × 81 mm (same as another type of square prism shaped artist's crayon), 8 × 11 × 81 mm (an extruded rectangle profile we thought might be advantageous). A tablet pen and the short pen from the toy slate were available for comparison.

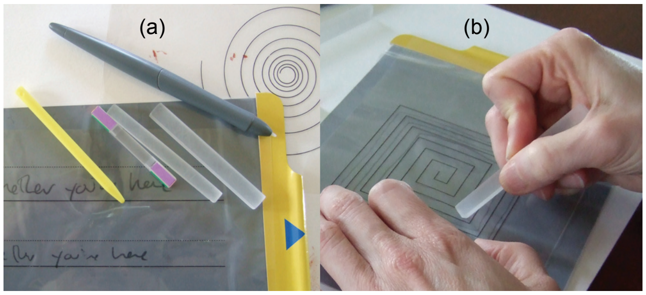

Figure 2. Interviews: (a) Conté mock-ups; (b) simulating fundamental tasks with the Magic Drawing Slate.

Primary observations are below (i.e. **1**a references task **1**):

1a. They use the corner, the end edge, and the side, and they blend with both non-dominant fingers and their dominant hand (by tucking the stick in their palm).

1b. Two different grips are used: a precise dynamic tripod grip, and a finger and thumb grip. The latter enables more contact surfaces to be accessed quickly.

1c. The tip becomes rounded with use, but artists can "find a corner" by feel as they rotate the shaft.

2a. Writing and tracing with a corner was similar to a pen, but the longer 81 mm mock-ups were preferred. Writing with Conté was less comfortable than a pen, but not especially uncomfortable either.

2b. The extruded rectangle profile could be held more securely, and did not reduce manipulative capability.

3a. "Tucking" Conté in the palm was easier than the tablet pen and up to three fingers could be used for touch.

3b. Transitioning contacts was fine, but took longer if the grip also had to be changed or re-adjusted. For example, changing from corner to side, or end to end.

3c. The most difficult contact to hold was a long side edge.

While clearly not a formal design study, these are useful general observations which we refer to later. Moreover, the open structure may have revealed aspects otherwise missed, and participants actually said it was fun.

CONTÉ CHARACTERISTICS AND DESIGN SPACE

In spite of its simple form factor, digital Conté has the potential to quickly switch between expressive input modes while providing additional parameters such as azimuth angle (barrel rotation), elevation angle (tilt), and pressure. After discussing the characteristics and capabilities of Conté, we outline a design space with includes using Conté alone and in combination with multi-touch.

Characteristics

The discussion here stems from our own observations supported by the artist interviews. Conducting formal quantitative experiments would be premature; we will discuss potential human factors studies as future work. To ground our discussion, we focus on the extruded rectangle Conté shape. The shape is pen-like given its crayon heritage, though compared to a pen it is shorter, faceted, and without a well-defined nib. Of course there will be some reduction in comfort and precision as the form factor deviates from a standard pen, and likewise a reduction in the number of stable contacts as the shape deviates from a ToolStone block. Our intention was to optimize this trade-off and we note that even efforts to establish an ideal pen shape are conflicted due to influences of individual preference and task [10,28]. The extruded rectangle shape was favoured by our interview participants and the slight irregularity should help users (and software) distinguish end edges and side faces. This form factor potentially supports 26 *different* contacts, classified into 7 *types* (Figure 3): 8 corners, 4 short end edges, 4 medium end edges, 4 long side edges, 2 end faces, 2 thick side faces, and 2 thin side faces.

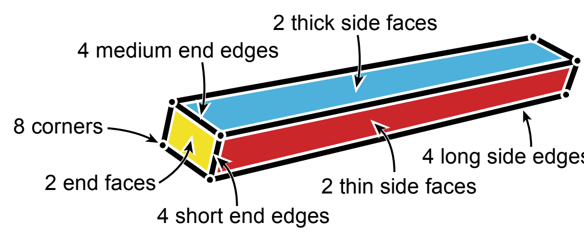

Figure 3. Seven types of contacts.

Contact Point Characteristics. For each type of contact, the combination of hand grip, contact shape, and equilibrium resulting from operating position dictate the availability of additional input parameters, level of precision, manipulation capability, and ability to maintain state when released (Figure 4). Song et al.'s [25] observation that people adopt different grips motivates a pen design which changes mode by sensing the current grip. With Conté, adopting different grips is a natural part of changing the desired contact.

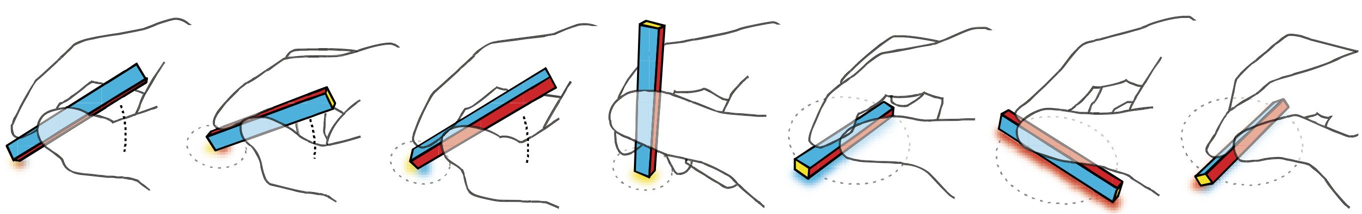

Figure 4. Typical hand grips used for different Conté contacts.

When contacting a corner, Conté is held like a pen using a precision grip, typically a dynamic tripod. The stick must be held at an angle to clearly disambiguate between adjacent contact points, so it has an unstable equilibrium (it must be supported to maintain a corner contact). Depending on the device thresholds for detecting adjacent contacts, usable elevation angles are less than a pen (well within 0 to 90°, clustered near 45°). There is no azimuth angle sensing limit, but in practice this is limited by wrist and elbow range-of-motion to about a 180° arc.

When contacting a short or medium edge, a slightly modified dynamic tripod grip [21] may be used and the increased contact area of the edge adds stability, though it must be held to maintain state. These factors reduce fine manipulation capability compared to the corner (or a pen), but the range of azimuth and elevation angles are similar.

In the exploratory interviews, we found that the long edge required a difficult thumb and index finger pinch grip. The small tolerance between the adjacent side faces reduces usable elevation angles to those near 45°. The instability of the grip and wrist and elbow range of motion reduce azimuth angles to approximately 180°, but rotating is difficult given the challenging grip.

The thick and thin side faces have a large contact surface area and low centre of gravity creating a very stable equilibrium: the grip can be loosened, or the stick set down, and the state maintained. The required pinch grip is much easier compared to the long edge. Also, the stability means azimuth angle covers the full 360° by manipulating with a clutching action. Elevation angle is invariant.

Contacting an end face suggests a further modified dynamic tripod grip which is moving towards a power grip [21]. Due to the high centre of gravity, the equilibrium is unstable and must be held to maintain state. Rolling the stick along the thumb enables 180° of azimuth rotation (like pen barrel rotation). Elevation angle is invariant.

Manipulation Characteristics. In our exploratory interview, participants commented on the extra time required to flip Conté end-over-end. This is similar to a pen nib-to-eraser transition, which takes 1.3 s on average [18]. With Conté, not all contact points are polar opposites; many nearby transitions should be quite fast: moving from a corner to an adjacent edge in a "rolling" adjacency *phrase* [5] for example. Hinckley et al. [15] also observed people "tucking" the pen in their palm to interleave dominant hand touch input. Based on our interviews, Conté's small size should make this even easier. The stable equilibrium of the side contacts also enable Conté to be set down and released (or grip partially loosened) yet maintain the same mode similar to ToolStone [23].

Cognitive Attributes. An open question is whether a user could recall command mappings for 26 different contact points, especially when coupled with rolling adjacency phrases and other parameter contexts such as azimuth angle. Similar strategies to those discussed by ToolStone's authors would increase usability, such as labelling faces for novices and adding tactile patterns for eye-free manipulation [23]. We have not explored this yet, but unlike ToolStone, Conté's greater variety of different contact points have diverse affordances which should assist recall. For example: the small corner size and tripod grip suggests drawing or writing; a medium edge is reminiscent of a highlighter and when oriented vertically, a text insertion point; a long side edge has a sharpness like a cutting blade; the flat ends act like stamps; and when laid on its side, Conté creates a tangible handle [16]. Also, unlike ToolStone, when Conté is used in the dominant hand on a direct input display it is the natural focus of attention. Thus, we can leverage proximal visual feedback to confirm the current mode. We display a rectangular "shadow" outlining the current contact and most activated modes have characteristic visual feedback such as displaying a guideline or rendering a dashed trail for lasso selection.

Design Space

The design space enabled by Conté is a super-set of the spaces demonstrated by commercial tablet pens, ToolStone [23], and Manual Deskterity [15]. At a base level, Conté can function like a pen, but also utilize the affordances and characteristics of different contacts to switch between modes (e.g. corner for drawing, end edge for highlighting).

Touch input can also be interleaved or coordinated with Conté input. For example, Conté could be used like ToolStone [14] in the non-dominant hand to set the mode for dominant hand touch – though contacts which use pen-like grips may be less effective [17]. Perhaps more interesting is *same-hand* touch input enabled by tucking Conté in the palm, setting it down on a stable side, or loosening the grip to free up a finger or thumb. The latter enables a hybrid style of "touch + handle" interaction, in which nearby touch input fine-tunes a mode enabled by Conté.

Conté can also realize the "pen + touch = new tools" design philosophy of Manual Deskterity. In addition to *touch context* techniques where the pen mode is inferred by the object context created by touch, Conté can be used for *pen context* techniques where the mode enabled by Conté contact creates the context for touch. Manual Deskterity's Tape Curve provides one example of this strategy where touch pins a pen stroke down to enter tape drawing mode. Since "the pen always writes" when used alone in Manual Deskterity, there are few possible *pen context* techniques — the only pen context is the stroke. This restrictive design rule makes sense with conventional pens. In contrast, Conté can enable multiple *"pen" context* techniques.

Below are primary qualities which define the total Conté design space covering these different functional styles:

- *Mode Mapping*. Consider precision and physical affordance when assigning contact point to mode (e.g. short edge is a highlighter, long side a ruler, end a stamp).
- *Group Modes*. Keep contact points of task related modes near each other (e.g. creation and revision at each end).
- *Use Adjacency Phrases*. Fast transitions among contacts can enable specialized modes (e.g. edge to corner roll).

- *Parameters for Direct Manipulation*. X-Y position and angle can be used for direct manipulation (e.g. position a guideline), to fine-tune a mode (e.g. pick default option), or to enable multiple modes per face like ToolStone.
- *Parameters at Invocation.* X-Y position and angle can fine-tune modes (e.g. object under contact to form context, angle on contact to pick sub mode).
- *Set It Down.* For stable sides, Conté will maintain the mode while freeing the hand for touch. Including nearby widgets revealed by the mode can further exploit this.
- *Use "Touch + Handle".* By loosening the grip on Conté, a finger or thumb can manipulate nearby widgets.
- *Leverage "Tucking."* Design techniques that interleave touch and Conté with one hand.
- *Set Context for Touch.* As discussed above, Conté's multimodal ability can enable "pen" context modes.
- *Non-Dominant Hand Usage.* Use more stable Conté contacts to set the mode for dominant hand touch.

In a later section, we demonstrate interaction techniques which encompass many of these qualities.

CONTÉ DEVICE PROTOTYPE

Creating a Conté device was challenging for multiple reasons. It was essential to make the device small, which ruled out some approaches. But we also added two more constraints: it must work with an unaltered multi-touch table, and it must be simple enough that researchers and tabletop hobbyists could create their own Conté sticks (we wanted to avoid complex electronics such as MTPen [25]). These constraints ruled out embedding multiple Anoto Pen sensors [4] or using magneto-electric coils like Toolstone [23] since they require too much space and tabletop alterations. Even if advanced magneto-electric pens such as the Wacom 6D Art Pen were compatible with direct touch input, two would have to be combined to support a two-ended Conté stick – and there is still the issue of altering the pressure-activated nib to work on corners, the limited range of elevation angles, and the size of the internal electronics. Instead, Conté emits and reflects infrared (IR) light which is captured by tabletop cameras, and translated by image processing algorithms into Conté input events. Emitting IR light uses simple and small electronics, and any diffuse infrared illumination multi-touch tabletop, such as the Microsoft Surface, can capture reflected IR patterns.

The simplicity of our design does make concessions: we detect and identify 10 out of the 26 possible contacts and we do not sense elevation angle. We detect a corner contact, but not *which* corner; likewise for short, medium, and long edges. Given the tiny surface area of corners and edges, there is little room to create a distinguishing ID pattern for image-based detection (consider that Surface ID tags are 19 × 19 mm). Since digital pens detect only one (or maybe two) contacts, this is still a significant increase and provides enough functionality to test Conté and begin exploring its potential. At the end of this paper, we discuss implementation ideas to detect all 26 contacts, but with added hardware complexity.

Hardware

Our prototype device is 9 × 11.5 × 84 mm and fits comfortably in the palm (Figure 5). It works by emitting IR light from each corner and reflecting IR light from paper labels affixed to the sides. Conceptually simple, we arrived at this design after much experimentation with materials and construction techniques. Two Osram SFH485 LEDs generate 880 nm near-IR light, matching the Surface IR pass band filter. Each LED is partially embedded in a highly polished 9 × 11.5 × 8 mm acrylic block using an open flame (Figure 5d). The acrylic block and LED are wrapped with foil tape to reflect IR light internally. Later, we cut small openings into each corner to let IR light escape (Figure 5c).

To house the electronics and fix the LED-embedded acrylic blocks at the tips, we cast everything in urethane resin using a custom mould (Figure 5b). This method creates a very solid feeling and durable stick, and since our mould was adjustable, it enabled fast prototyping of different shapes and versions. An alternative approach would be to mill a housing out of plastic, or print one using a solid printer, but these require more set up and would require high tolerances for the LED-embedded acrylic blocks.

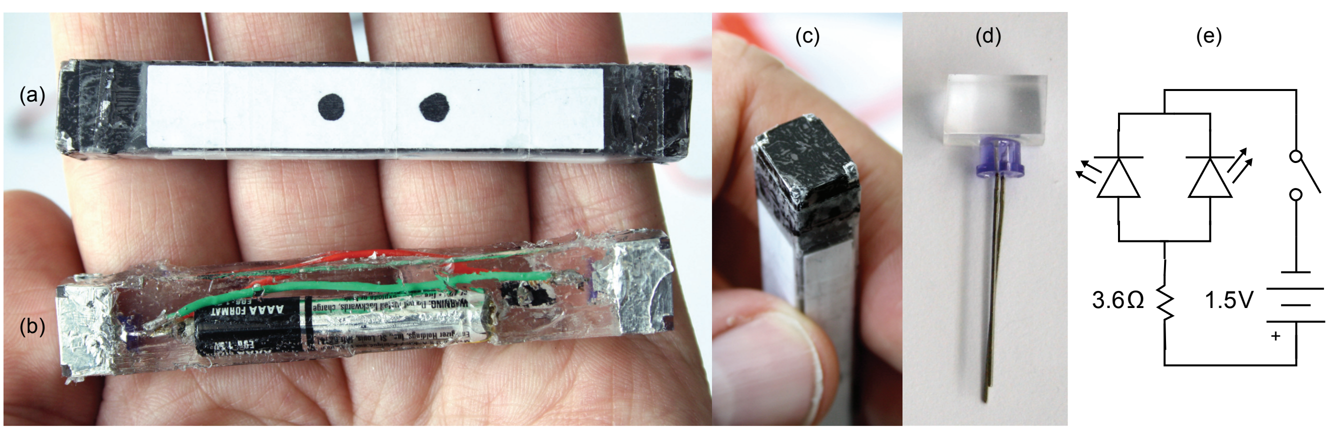

Figure 5. Infrared Conté prototype (shown actual size): (a) completed device; (b) after casting, showing battery and circuit inside; (c) end showing IR "windows"; (d) IR LED embedded in acrylic block; (e) circuit diagram.

After casting, the stick is sanded to fine-tune dimensions and painted matte black to increase the contrast of IR light. Then, small 2×2 mm openings are cut into each corner to emit IR light from the LEDs (Figure 5c). On one end, we paint a 3 mm white dot for identification. Finally, we glue foil-backed white paper labels to the side faces, and wrap everything in thin matte transparent tape. The foil backing increases IR reflectivity, and the tape prevents rubbing the paint off as well as protecting the Surface diffuser from scratches. Each label has 3 mm black dots: using the number of dots and size of the label, we can identify the side.

Our basic prototypes are powered by two external batteries supplying 3 V. In spite of being tethered by a thin wire, we found we could manipulate them surprisingly well. However, the protruding wire makes one side face unusable and does reduce manipulative capability somewhat. We found the Osram LEDs require a minimum voltage of 1.4 V, so they can be powered using a single alkaline AAAA battery. At just over 8 mm in diameter, we could embed one and keep a small form factor. With the LEDs wired in parallel and a 3.6 Ω resistor (Figure 5e), enough IR is emitted for tracking. But, compared to the 3 V tethered version, the emitted IR is lower making tracking high velocity movements less reliable (due to camera exposure blurring which we discuss in the next section). Our battery powered Contés operate for about 25 hours. The circuit draws 20 mA (due to battery voltage-current characteristics and LED forward current characteristics) and alkaline AAAA batteries are rated for 500 mAh. Rechargeable NiMH or NiCd AAAA batteries are only rated 1.2 V, so they would require an LED driver (such as the Zetex ZXSC300). We continue to investigate this rechargeable solution.

Unlike Hinckley et al.'s IR pen [15], we do not use tip switches to only activate the LEDs when touching the display. Attaching miniature switches to all contact points would be very difficult. Instead, we found that the combination of pinpoint IR light from the corners, and the Surface's high quality diffuser (Evonik ACRYLITE 7D006), resulted in only slight hover artefacts.

Software

Our software translates the IR light patterns of Conté into events describing the current contact point, position, and, if available, azimuth angle. It is written in C# using the Emgu 2.2.1 (www.emgu.com) OpenCV wrapper. Figure 6 illustrates the following image processing steps.

Pre-Processing. Using the Microsoft Surface SDK 1.0 SP1, we access the 768×576 px, 8 bit greyscale image taken by the internal IR cameras. A Top Hat morphological operation is applied using 3 iterations of a 3×3 structuring element. This brightens Conté's sharp, bright shapes and darkens the duller smooth contacts of fingers and palms. We found this initial operation to be extremely important.

Position Detection. Next, we locate the approximate size and position of the Conté contact. The pre-processed image is binarized using a *variable* threshold value and then 20 iterations of the Dilate morphological operation is applied, creating large connected blobs. The variable threshold is computed from the velocity of the Conté contact in the previous frame. Specifically, we linearly interpolate between a threshold of 20 when Conté is moving faster than 5 mm/s, and a threshold of 66 when Conté is moving slower than 1 mm/s. When at rest or moving slowly, the high threshold isolates Conté from everything else. With fast movements, the camera exposure blurs the image of the corners or sides, and the threshold must be reduced. Of course, as the threshold is reduced, the intensity of other types of surface contacts will be above the threshold. To address this, we use the binarized image blob closest to the last Conté position — a surprisingly effective and simple rule. Using this approximate contact position, we can update the velocity (which is low-pass filtered with a 0.03 Hz cut off). If the velocity is above 1.4 mm/s, we stop processing and the approximate contact position, together with the last known Conté contact point and azimuth angle, are used to construct a Conté movement event.

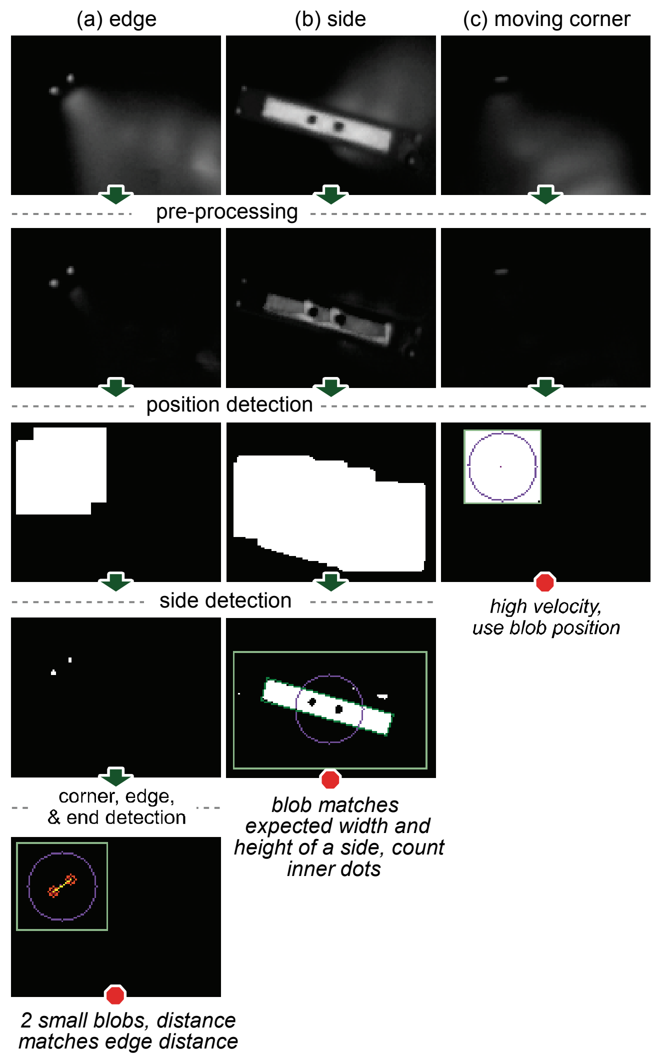

Figure 6. Image processing examples: (a) edge; (b) side; (c) moving corner. Rows show results of an image processing step: capture; pre-processing; position detection; side detection; corner, edge, and end detection.

Side Detection. If the velocity is lower, we continue processing and check whether a side face is touching. The pre-processed image is first binarized using a threshold of 66. Then, the minimum area rectangles of white outer connected component blobs are found (using Emgu `FindContours` and `GetMinAreaRect`). These are compared to the expected width and height of the thick or thin side face labels. If a match is found, the number of black connected component blobs within the matched blob is used to identify *which* side is touching. The minimum area rectangle also provides the azimuth angle of the side.

Corner, Edge, and End Detection. If no matching blobs were found in the side detection step, we search for corners. First, the pre-processed image is binarized using a threshold of 75. Any connected component blobs with an area greater than 16 pixels are removed. The number of remaining blobs and their relative distance determines the contact point: 1 blob is a corner; 2 blobs are an edge if their distance is ± 3 px of 9.5 px for short, 14 px for medium, 100 px for long; and 4 blobs is an end. Any other number of blobs is considered ambiguous and flagged as such. If an end is detected, we apply a lower threshold of 55 to the area between LEDs and determine if there is a white dot which identifies which end is contacting. An improved strategy would be to find the *best* match to all known contacts. For example, 3 blobs are either a corner with the stick held high, or an end with a slight tilt. These could be identified using intra-blob distances, rather than labelled ambiguous.

Performance. The main optimization we made was to restrict the processing area in the *Side Detection* and *Corner, Edge, and End Detection* steps to the contact blob bounding rectangle found in the *Rough Position Detection* step. Our algorithm runs at 45 Hz on a standard Microsoft Surface (Vista, 2.13 GHz dual core CPU, 2 GB RAM).

INTERACTION TECHNIQUE TEST-BED

An obvious application of digital Conté is an artist's sketching application which mimics how a real Conté crayon creates different line thicknesses. We go beyond this direct analogy and demonstrate the potential of Conté more fully in a sketching and drafting inspired application built using C#, WPF, and the Microsoft Surface SDK. Our application has enough complexity to force techniques to be mutually compatible and useful. It includes sketch-based drawing annotation like Manual Deskterity [15], but also demonstrates additional types of precise pen-like input and modal tools like guidelines. Earlier we discussed how Conté can support Manual Deskterity's *touch context* interactions, but we focus on techniques which set Conté apart: unimanual multi-modal input leveraging the characteristics of different contact points and *pen context* techniques where Conté sets the context for touch.

Pen-Like Input

The level of precision and affordance when contacting the corner or end edges suggest one-handed pen-like interactions. The corner is used for freehand drawing, since this is arguably the most precise pen-like contact (Figure 7a). Strokes made with the short end edge are interpreted as shape gestures, a simple solution enabled by Conté for the "ink vs. gesturing" problem [18]. These shape strokes are analyzed with the .NET 4 ink recognizer and replaced by a beautified version (Figure 7b). For example, a circular shape is recognized and replaced with a perfect circle; likewise for ellipses, rectangles, squares, triangles, etc. We use a dashed stroke pattern in the current stroke colour for visual feedback. The medium end edge performs a lasso selection (Figure 7d), a third pen-like input mode. Lasso selections are less precise than drawing and writing, but still require control over shape. A thick black dashed stroke provides visual feedback.

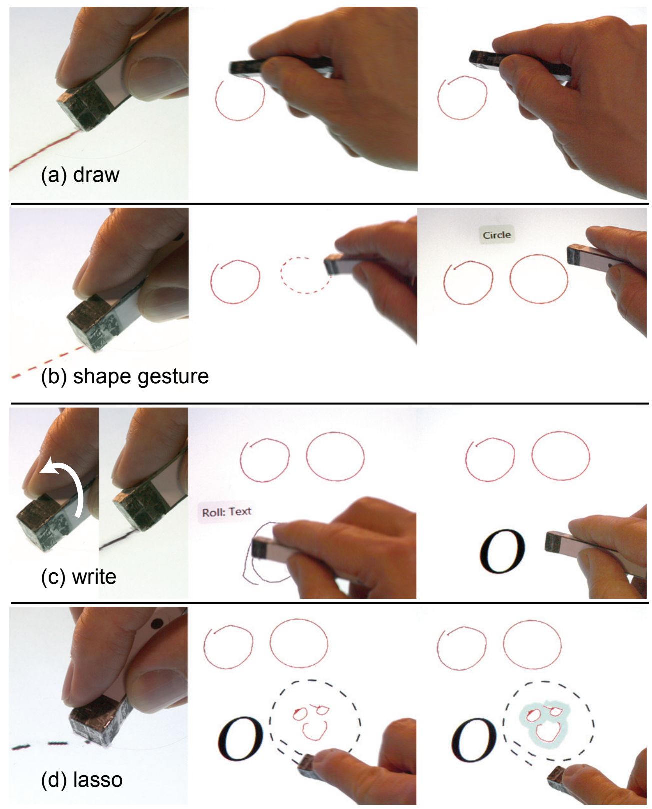

Figure 7. One handed pen-like input modes using corner and end edges: (a) freehand drawing and annotating with the corner; (b) making shape recognition gestures with the short edge; (c) a "roll" from short edge to corner changes corner mode to writing with text recognition; (d) medium edge for lasso selection.

In support of our drafting application scenario, users can enter typographic text by writing with Conté. Like drawing, writing is a precise task requiring fine grain manipulation, so a corner contact is most appropriate. Automatically distinguishing between drawing and writing is unreliable due to many ambiguous strokes (consider strokes resembling an 'O', 'l', or 'L' for example). With Conté, the user explicitly switches between writing and drawing mode. If our prototype Conté device could distinguish between corners, two different corners could be used. Instead, we exploit contact point adjacencies, and use a "roll" from a short end edge to a corner to enter a corner writing mode. A roll is recognized when there is a change to a corner contact less than 1 s after a short end edge contact. We provide visual feedback with a little notification tab which says "Roll: Text" and the stroke colour changes to the current font colour. Corner strokes return to freehand drawing after a subsequent mode change, or when the user explicitly exits with another short end edge to a corner "roll."

Contextual Commands

Contextual commands are common in most applications. In Windows, these are accessed with a right-click on an object and selecting from a context menu (such as 'copy'). In Deskterity, copying is performed by dragging the pen off an object held by a finger. This is an elegant solution, but requires two hands and makes multiple and distant duplication difficult. Moreover, adding more contextual commands means adding more "touch + pen" gestures.

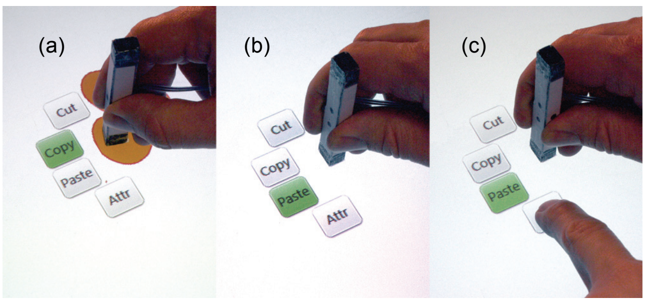

Figure 8. Contextual commands using the end contact: (a) a menu appears with 'copy' as default action since the thick side faces up-down; (b) 'paste' is default when the thick side faces left-right; (c) finger selects other commands which also become the new default for the current thick side orientation.

With Conté, an end contact on an object opens a contextual menu (Figure 8c) showing 'cut', 'copy', 'paste', and 'attr' (paste clipboard object *attributes* only, such as colours, typeface, etc.). The end stamping motion affordance seems to match these actions. To support one-handed operation, we use azimuth angle to *pre-select* a default command when the stick is immediately lifted. For example, when the thick side faces left-right, 'paste' is pre-selected, but when facing up-down, 'copy' (Figure 8a,b). The menu is only revealed after 200 ms, encouraging expert users to quickly access these default commands without visual clutter and enabling novice-to-expert transition. As a further refinement, when a command is selected by a non-dominant finger, it is used as the new default action for the current Conté azimuth orientation. Thus, 'attr' could become the new default when the thick side faces left-right, allowing the user to rapidly paste attributes to multiple drawing objects. Although not implemented, a different context menu could be associated with each end of the stick.

Attribute Palettes

Laying Conté down on one thick side opens one of four attribute palettes (fill colour, stroke colour, font colour, font properties). Since this a very stable contact, Conté can be released (Figure 9c) and an attribute selected with the same hand, or held as the non-dominant hand selects. We placed two touch sensitive buttons just above the stick to cycle through different palettes. With this placement, single-handed, simultaneous Conté and touch manipulation is possible by loosening the grip and using the middle and ring fingers to tap (Figure 9a). By touch-dragging on the palette bezel (Figure 9b), the palette may be "peeled" off Conté to remain visible after Conté is lifted. This can be done with non-dominant fingers or with the dominant hand by loosening the grip, dropping the index finger, and then tucking Conté in the hand. When cycling palettes, peeled palettes are temporarily brought to Conté's location with their peeled location shown as a dashed outline. Palettes are hidden by tapping the same thick side on or near the palette. This has the feeling of "picking up" the palette. We found that with palettes it can be natural to pass Conté to the non-dominant hand similar to ToolStone [23], and the dominant hand is free to touch for attribute selection.

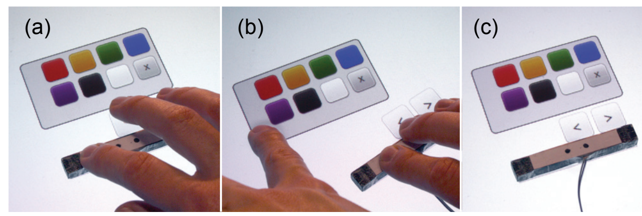

Figure 9. Attribute palettes using a thick side face: (a) one-handed cycling through palettes; (b) "peeling" a palette off to keep it visible; (c) side face is stable, so palette mode is held when Conté is released.

We explored alternate methods of cycling palettes such as using an up or down rolling action from side to adjacent long edge. Theoretically interesting, in practice it was difficult to maintain the necessary grip (partly because of the smooth sides created by wrapping Conté in tape).

Guidelines and Alignment

In commercial applications, guidelines are usually created by dragging off rulers anchored on the edge of the canvas. On a large tabletop this requires reaching, clutters the drawing area, and favours horizontal and vertical guidelines.

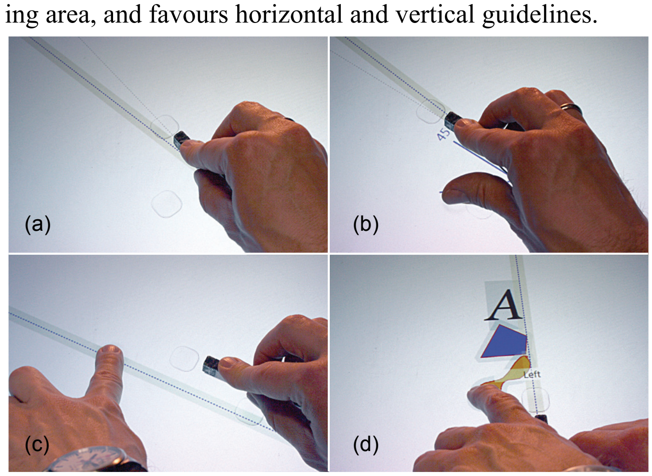

Figure 10. Guidelines using thin side face: (a) revealing and positioning temporary guideline with Conté; (b) adjusting snap angle with thumb; (c) "peeling" temporary guideline to make it permanent; (d) alignment tool, in this case left-aligning objects with a left-swipe.

With Conté, users can create guidelines at any angle by contacting the thin side face (Figure 10a). The stick translates and rotates the guideline, and similar to palettes, they may be peeled off (or perhaps "pinned down") with a dominant or non-dominant touch (Figure 10c). Guidelines are "picked up" by contacting the thin side nearby in the same orientation. Adjusting the guideline snap angle is another example of single-handed, simultaneous Conté and touch manipulation: with a loosened grip, the thumb is free to adjust snap angle by dragging (Figure 10b).

The guideline tool also aligns objects. Tapping or swiping a touch sensitive target just above Conté aligns currently selected objects. Swiping left left-aligns, swiping right right-aligns, and tapping centre-aligns (Figure 10d). While holding Conté, these are most comfortably done with the non-dominant hand, though centre taps are not too difficult with the dominant hand. However, given the stable thin side, the dominant hand can perform a complete alignment task: after positioning a temporary alignment guideline, Conté is released, and still in guideline mode the desired alignment command performed with a tap or swipe, and then the stick picked up to remove the temporary guideline.

Mouse-Like Pointing

Rekimoto and Sciammarella [23] pondered the idea of adding a button to ToolStone to use it as a mouse, and Forlines at al. [9] found that using a conventional mouse on a tabletop can outperform touch for some tasks. Inspired by this, when Conté is laid down on one thick side face it behaves like a standard mouse as it controls an arrow cursor. A stylized image of a mouse is rendered around Conté with its entire surface acting as a single touch sensitive button. This large button accommodates the restricted free finger movements when "clicking," while the remaining fingers continue to grip Conté. With a little practice, we could click and drag objects while maintaining a grip. We create a "mouse-aligned reference frame" [13] by mapping Conté movement vectors in display space to cursor movements. A pointer acceleration function is tuned for aggressive cursor movement with fast movements, but near 1:1 control-display gain when moved slowly. This enables precise selection and minimizes clutching over long distances [6].

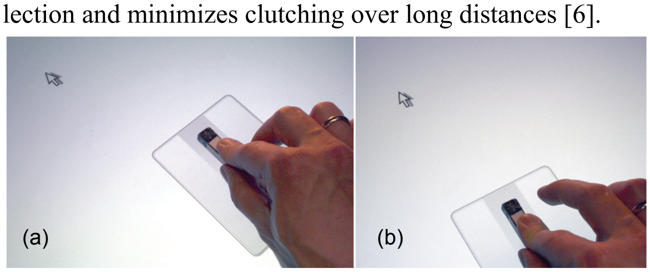

Figure 11. Mouse-like pointing: (a) point; (b) click.

CONCLUSION AND FUTURE WORK

Within the simple idea of recognizing different contacts of a faceted drawing stick lie hardware and software challenges, a new interaction design space, and new human factors questions to investigate. The work discussed in this paper focuses on the first two of these challenges to enable and motivate the third.

Focusing on IR hardware as we have has the advantage of fabrication tractability and tabletop compatibility, but removing these self-imposed constraints enables alternate strategies. One improvement we plan to test is adding a contact microphone to sense movement over the diffuser. If the sound profiles are distinct, this could act like a primitive tip switch. With a switchable diffuser [14] and colour-coded Conté faces, all 26 contacts can be recognized with basic image processing and a model of contact colour patterns. We have already implemented this strategy using a clear tabletop for a future quantitative manipulation experiment. By accepting more hardware complexity, accelerometers and/or a gyroscope could be embedded to accurately measure orientation, and the contact inferred similar to ToolStone. Diffuse or emitted IR light may continue to provide X-Y position information, or for a capacitive touch device, an outer layer of conductive material could be used.

Enabled by a hardware solution, and motivated by demonstrations of an interaction design space, we can turn to human factors. The many characteristics of Conté must be verified and quantified, and we are already planning an investigation of contact-to-contact transitions. Like Conté itself, this appears straightforward, but closer analysis reveals 54 different transitions (accounting for symmetry) which need to be contrasted and compared. This will complement Li et al.'s [18] work and help situate Conté among other mode switching techniques. Other investigations include precision, angular manipulation, grip styles, and recall capability: can people use 26 or more mappings? If not, how many? Related are ergonomic studies to test alternate form factors, construction materials, and tactile patterns.

The interaction techniques described here demonstrate key qualities of the Conté design space, but they may only hint at the potential. For example, some qualities can be pushed farther, such as leveraging "tucking," and others we have not addressed directly like switching Conté between hands. Art historians have argued that the invention of the versatile Conté crayon influenced a new style of art [24]. At the risk of sounding overly enthusiastic, perhaps digital Conté could have a similar effect on human-computer interaction.

ACKNOWLEDGEMENTS

This work was partially supported by the SHIVA project (Interreg IV A −2 seas) and FEDER.

REFERENCES

1. Benko, H., Wilson, A.D., and Baudisch, P. Precise selection techniques for multi-touch screens. *Proc. of the SIGCHI conference on Human Factors in computing systems*, ACM (2006), 1263-1272.
2. Bi, X., Moscovich, T., Ramos, G., Balakrishnan, R., and Hinckley, K. An exploration of pen rolling for pen-based interaction. *Proc. of the 21st annual ACM symposium on User interface software and technology*, ACM (2008), 191-200.
3. Brandl, P., Forlines, C., Wigdor, D., Haller, M., and Shen, C. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. *Proc. of the working conference on Advanced visual interfaces*, ACM (2008), 154-161.
4. Buxton, W.A.S. Chunking and phrasing and the design of human-computer dialogues. In *Human-computer interaction: toward the year 2000*. Morgan Kaufmann Publishers Inc, 1995, 494-499.
5. Casiez, G., Vogel, D., Balakrishnan, R., and Cockburn, A. The Impact of Control-Display Gain on User Performance in Pointing Tasks. *Human-Computer Interaction 23*, 3 (2008), 215-250.
6. Deming, K. and Lank, E. *Mode Selection Techniques for Pen Input Systems*. San Francisco State University, 2005.
7. Fitzmaurice, G., Baudel, T., Kurtenbach, G., and Buxton, B. A GUI paradigm using tablets, two-hands and transparency. *CHI '97 extended abstracts on Human factors in computing systems: looking to the future*, ACM (1997), 212–213.
8. Forlines, C., Wigdor, D., Shen, C., and Balakrishnan, R. Direct-touch vs. mouse input for tabletop displays. *Proc. of the SIGCHI conference on Human factors in computing systems*, ACM (2007), 647–656.
9. Frisch, M., Heydekorn, J., and Dachselt, R. Investigating multi-touch and pen gestures for diagram editing on interactive surfaces. *Proc. of the ACM International Conference on Interactive Tabletops and Surfaces*, ACM (2009), 149–156.
10. Goonetilleke, R.S., Hoffmann, E.R., and Luximon, A. Effects of pen design on drawing and writing performance. *Applied Ergonomics 40*, 2 (2009), 292-301.
11. Greenberg, S. and Buxton, B. Usability evaluation considered harmful (some of the time). *Proc. of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ACM (2008), 111–120.
12. Guiard, Y. Asymmetric division of labor in human skilled bimanual action: the kinematic chain as a model. *Journal of Motor Behavior 19*, 4 (1987), 486-517.
13. Hartmann, B., Morris, M.R., Benko, H., and Wilson, A.D. Augmenting interactive tables with mice & keyboards. *Proc. of the 22nd annual ACM symposium on User interface software and technology*, ACM (2009), 149–152.
14. Hilliges, O., Izadi, S., Wilson, A.D., Hodges, S., Garcia-Mendoza, A., and Butz, A. Interactions in the air: adding further depth to interactive tabletops. *Proc. of the 22nd annual ACM symposium on User interface software and technology*, ACM (2009), 139-148.
15. Hinckley, K., Yatani, K., Pahud, M., et al. Pen + touch = new tools. *Proc. of the 23nd annual ACM symposium on User interface software and technology*, ACM (2010), 27–36.
16. Jordà, S., Geiger, G., Alonso, M., and Kaltenbrunner, M. The reacTable: exploring the synergy between live music performance and tabletop tangible interfaces. *Proc. of the 1st international conference on Tangible and embedded interaction*, ACM (2007), 139–146.
17. Kabbash, P., MacKenzie, I.S., and Buxton, W. Human performance using computer input devices in the preferred and non-preferred hands. *Proc. of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, ACM (1993), 474-481.
18. Li, Y., Hinckley, K., Guan, Z., and Landay, J.A. Experimental analysis of mode switching techniques in pen-based user interfaces. *Proc. of the SIGCHI conference on Human factors in computing systems*, ACM (2005), 461-470.
19. Lin, J., Newman, M.W., Hong, J.I., and Landay, J.A. DENIM: finding a tighter fit between tools and practice for Web site design. *Proc. of the SIGCHI conference on Human factors in computing systems*, ACM (2000), 510-517.
20. Moran, T.P., Chiu, P., and Melle, W. van. Pen-based interaction techniques for organizing material on an electronic whiteboard. *Proc. of the 10th annual ACM symposium on User interface software and technology*, ACM (1997), 45-54.
21. Napier, J.R. The prehensile movements of the human hand. *The Journal of Bone and Joint Surgery. British Volume 38-B*, 4 (1956), 902-913.
22. Ramos, G., Boulos, M., and Balakrishnan, R. Pressure widgets. *Proc. of the SIGCHI conference on Human factors in computing systems*, ACM (2004), 487-494.
23. Rekimoto, J. and Sciammarella, E. ToolStone: effective use of the physical manipulation vocabularies of input devices. *Proc. of the 13th annual ACM symposium on User interface software and technology*, ACM (2000), 109-117.
24. Scheyer, E. French Drawings of the Great Revolution and the Napoleonic Era. *The Art Quarterly IV*, 4 (1941), 187-204.
25. Song, H., Benko, H., Guimbretière, F., Izadi, S., and Cao, X. Grips and gestures on a multi-touch pen. *Proc. of the SIGCHI conference on Human factors in computing systems*, ACM (2011).
26. Tian, F., Xu, L., Wang, H., et al. Tilt menu: using the 3D orientation information of pen devices to extend the selection capability of pen-based user interfaces. *Proc. of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ACM (2008), 1371-1380.
27. Vogel, D. and Baudisch, P. Shift: a technique for operating pen-based interfaces using touch. *Proc. of the SIGCHI conference on Human factors in computing systems*, ACM (2007), 657-666.
28. Wu, F. G. and Luo, S. Design and evaluation approach for increasing stability and performance of touch pens in screen handwriting tasks. *Applied Ergonomics [Kidlington] 37*, 3 (2006).
29. Wu, M., Shen, C., Ryall, K., Forlines, C., and Balakrishnan, R. Gesture Registration, Relaxation, and Reuse for Multi-Point Direct-Touch Surfaces. *Proc. of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, IEEE Computer Society (2006), 185–192.
30. Yee, K-P. Two-handed interaction on a tablet display. *CHI '04 extended abstracts on Human factors in computing systems*, ACM (2004), 1493-1496.
31. Computer Sketchpad. *Science Reporter (Television Series)*, 1964.