Sensing Techniques for Tablet+Stylus Interaction

Ken Hinckley[1], Michel Pahud[1], Hrvoje Benko[1], Pourang Irani[1,2], Francois Guimbretiere[1,3], Marcel Gavriliu[1], Xiang 'Anthony' Chen[1], Fabrice Matulic[1], Bill Buxton[1], & Andy Wilson[1]

[1]Microsoft Research, Redmond, WA
{kenh, benko, mpahud, bibuxton, awilson}@microsoft.com

[2]University of Manitoba, Dept. of Comp. Sci., Winnipeg, MB, Canada, irani@cs.umanitoba.ca

[3]Cornell University, Information Science Department, Ithaca, NY
francois@cs.cornell.edu

ABSTRACT

We explore *grip* and *motion* sensing to afford new techniques that leverage how users naturally manipulate tablet and stylus devices during pen + touch interaction. We can detect whether the user holds the pen in a writing grip or tucked between his fingers. We can distinguish bare-handed inputs, such as drag and pinch gestures produced by the nonpreferred hand, from touch gestures produced by the hand holding the pen, which necessarily impart a detectable motion signal to the stylus. We can sense which hand grips the tablet, and determine the screen's relative orientation to the pen. By selectively combining these signals and using them to complement one another, we can tailor interaction to the context, such as by ignoring unintentional touch inputs while writing, or supporting contextually-appropriate tools such as a magnifier for detailed stroke work that appears when the user pinches with the pen tucked between his fingers. These and other techniques can be used to impart new, previously unanticipated subtleties to pen + touch interaction on tablets.

Author Keywords

Sensing; pen+touch; motion; grip; tablet; bimanual input

ACM Classification Keywords

H.5.2 [Information Interfaces and Presentation]: Input

INTRODUCTION

In the modern context of multi-touch tablets, pen input has received renewed interest, particularly when used as a distinct modality in tandem with touch [21,34,55]. However, many problems and ambiguities arise because the system can't easily determine whether the user intends to interact with the pen, with touch, or with the two in combination. As one example, systems employ various heuristics to ignore *unintentional* touch inputs associated with resting the palm on the screen while writing, but typically these come at the expense of limiting simultaneous *intentional* touches produced by the nonpreferred hand [1,2,42].

UIST '14, October 05 - 08 2014, Honolulu, HI, USA

ACM 978-1-4503-3069-5/14/10…$15.00.
http://dx.doi.org/10.1145/2642918.2647379

As witnessed by the proliferation in mobile sensing [7,12,19,23,24,39], there is great potential to resolve such ambiguities using sensors, rather than foisting complexity on the user to establish the missing context [6]. As sensors and computation migrate into tiny mobile devices, pens no longer need to be conceived primarily as passive intermediaries without a life once they move away from the display.

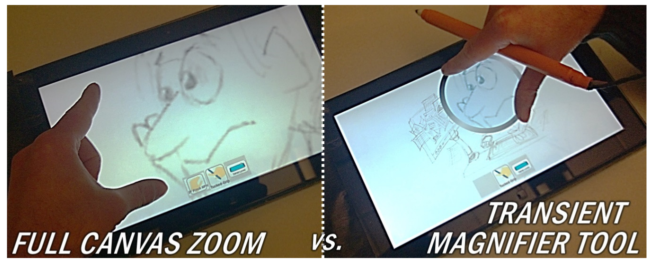

Figure 1. Our sensors can distinguish bare-handed (*left*) vs. pen-in-hand (*right*) inputs to evoke contextually-appropriate tools.

To extend this perspective, we show novel ways of evoking contextually-appropriate tools. For example, if the user tucks the pen while performing a pinch gesture (*Figure 1, right*), this brings up a magnifying tool particularly well suited to detail work with the pen. This is sensed by detecting the *Tuck* grip in combination with a hard-contact motion signal on the pen *at the same time* the hand holding it contacts the touchscreen. Performing a bare-handed pinch gesture with the other hand, however, does not produce this pattern and can therefore invoke a distinct behavior (*Figure 1, left*).

We also report observations of a rich vocabulary of naturally occurring behaviors, including *tuck* vs. *palm* grips for stowing the pen, as well as a new class of finger *extension grips* (from the hand holding the pen) for touchscreen manipulation. That such behaviors have gone largely unnoticed, much less actively sensed by pens and tablets, reflects the opportunities for contextual adaptation and fresh nuances of expression afforded by adding a few new sensors.

Overall, our contributions include the following:

- Observations of natural grips and usage patterns that arise when using pen and touch together on a tablet.
- Novel hardware / software solutions that sense grip and motion to capture the full context of stylus and tablet use.
- Using sensors to *mitigate unintentional touch* (from the palm, or from the thumb when picking up the tablet), but also to *promote intentional touch* by the nonpreferred hand, or via extension grips to interleave pen and touch.
- Novel contextually-appropriate tools that combine grip, motion, and touchscreen contact, including distinct tools

for bare-handed input, pinch input while tucking the pen, and drafting tools that the user can summon with the nonpreferred hand when the pen is poised for writing.
- Preliminary observations and user reactions arising from initial user experience with these tools and techniques.

Collectively, these observations and novel techniques illustrate how contextual sensors on modern stylus and tablet devices have the potential to ameliorate ambiguities inherent in simultaneous pen+touch interaction, while also affording a broad design space of novel interactions.

RELATED WORK

Our research threads together themes from mobile sensing, sensor-augmented pens, grip sensing, and pen + touch input.

Mobile Sensing on Handhelds

Tilt, pressure, and proximity sensing on mobiles enables contextual adaptations such as detecting handedness [14], portrait/landscape detection [19], or walking vs. stationary usage [40]. Grip sensing allows a mobile to detect how the user holds it [53], or to use grasp to automatically engage functions [49] such as placing a call, taking a picture, or watching a video [24]. We adopt this perspective of sensing natural user behavior, but bring it to pen and tablet interaction, where the behaviors to sense—and the new possibilities for pen+touch—remain largely unexplored.

Multi-touch input and inertial sensors (IMU's with 3-axis gyroscopes, accelerometers, and magnetometers) afford new possibilities for mobiles to discern user intent based on grasp and motion dynamics [3]. Devices can distinguish if the user touches the screen with the thumb vs. the index finger [12], or use grip sensing to avoid accidental screen auto-rotation [7]. Whacking [23] or jerking motions [39] offer out-of-band signals distinct from incidental handling of a device. We illustrate new techniques that leverage these types of motion sensing, grip sensing, and multi-touch inputs when they are distributed across separate pen and tablet devices.

Grips and Sensing for Tablets

Lightweight tablets afford many new grips, movements, and sensing techniques. BiTouch and BiPad [52] demonstrates bimanual interactions that take natural tablet grips into account. Sensing tablet grips affords bringing up a touchscreen keyboard in a contextually-appropriate location [8]. GroupTogether [32] senses clusters of co-located users to support lightweight exchange of information by subtly orienting a tablet towards another user with micro-mobility gestures [29]. Our system is the first to implement full grip sensing and motion sensing—on both tablet and stylus at the same time—for pen + touch interaction.

Palm Detection and Unintentional Touch Handling

Palm contact can cause false activations during pen + touch interaction [1,2,42]. For example, many iPad note-taking applications include palm-blocking (e.g. [11,35,36]), but details of these heuristics are unpublished and vary widely; they often appear to rely on application-specific assumptions about how and where the user will write. Other palm-rejection techniques require the user to bring the pen tip on or near the screen before setting the palm down, which requires users to modify their natural movements (e.g. [55], or Microsoft Windows tablets, which turn off touch when the pen comes in range). Our approach uses sensors to detect when a touchscreen contact is associated with the hand holding the pen, thus showing that the problem can be framed as one of sensing appropriate contextual information.

Sensor-Augmented and Multi-DOF Stylus Input

Auxiliary tilt, roll, and other stylus degrees-of-freedom can be combined [15] to call up menus [50] or trigger mode switches [4] without necessarily disrupting natural use [54]. Conte [51] explores a tilt-sensitive "digital Conte crayon" for pen + touch interaction on a tabletop. In particular, Conte explores interactions where the user orients or holds or the crayon on the display, while other fingers from the same hand reach out and touch the screen to pick colors or choose modes. We consider similar capabilities where the user can extend one or more fingers while tucking the stylus. We also sense distinct contexts with separate functions, even when the user holds the pen away from the screen.

Pens can be augmented with motion [17,48], grip [46,47], and near-surface range sensing [28]. MTPen [46] uses grip sensing to detect a tripod writing grip, or to invoke different brushes. Sun et al. [47] use an integrated IMU on the pen to assist grip recognition, an approach which we adopt here, and also sense the orientation of the tablet (e.g. horizontal vs. drafting table). Context sensing examples have also been proposed, such as sensing if the user forgets the pen [17]. Lee et al. [26] employ an orientation-sensing stylus to reduce occlusion problems. Our work goes beyond these efforts; compared to the most closely related work (i.e. MTPen [46], FlexAura [28], and [17,21]), the main things that distinguish our contribution are our behavioral observations, the holistic pen and tablet sensing platform, and application of new techniques to well-considered UI challenges.

Synchronous Gestures and Cross-Channel Inputs

The literature shows examples of distributed sensing techniques such as bumping two tablets together [16] or tapping a mobile phone against a multi-touch tabletop [41]. Synchronous events or correlations across input channels can also enable new types of gestures. The 1Line keyboard [9] uses a bezel thump as a spacebar by noting that normal keystrokes produce a bump at the same time as the touch-down event, whereas a bezel thump arrives without a corresponding touch event. Smart-Its Friends [22] pairs two devices when the user shakes them together. *Are You With Me?* [27] uses motion signal similarities to sense if two mobile devices are being carried by the same person, while *Your Phone or Mine?* [38] correlates mobile phone motions with depth image patches. Sensor Synaesthesia [20] and GripSense [12] combine touchscreen and motion signals. Our research contributes cross-device and cross-channel inputs for a distributed tablet–stylus sensing system to detect unique contexts and behaviors that arise in pen+touch input.

OBSERVATIONS OF PEN & TABLET BEHAVIORS

Several papers identify distinct grips for tasks such as writing and brushing [46,47], or holding tablets [8,52]. But the literature still lacks an inventory of pen grips and postures while interacting on a tablet with pen + touch. We therefore sought to enumerate common grips that arise in digital pen-and-tablet tasks, and particularly touchscreen interactions articulated while the pen is in hand. Our task thus spans note-taking with a pen, as well as a secondary task (web browsing) that primarily employs touch. We chose web browsing for this secondary task because writing side-by-side with reading the web offers a compelling scenario from the literature for digital ink (e.g. [18,44]).

Nine people (4 female, 1 left-handed) participated in the observational study. Users interacted with a 10” Windows 8 tablet, with multi-touch and an electromagnetic pen digitizer, using OneNote to jot notes and Internet Explorer to browse the web. We observed a wide variety of behaviors (B1-B12) that helped inform our designs, as well as many previously unrecognized pen grips, as illustrated in Figure 2 (some are drawn from the literature). At present, we focus on behaviors of right-handers; left-handers are known to exhibit a variety of additional grips and accommodations (e.g. [13]).

B1. Stowing the pen during touch. Users often stowed the pen [21,46] when performing touch gestures. Users only put the pen down when they anticipated they wouldn’t need it again for a prolonged time, or if they encountered a task that they felt was too awkward to perform with the pen in-hand, such as typing a lot of text using the on-screen keyboard.

B2. Tuck vs. Palm for stowing the pen. Unlike previous studies, we observed two distinct grips that users employ to stow the pen. All users exhibited the *Tuck* grip (pen laced between fingers), but many also employed a *Palm* grip (with fingers wrapped lightly around the barrel) to stow the pen during pauses or to afford touch interactions.

B3. Preferred pen stowing grip depends on task context. For users that employed both *Tuck* and *Palm* grips, *Tuck* afforded quick, transient touch interactions, while *Palm* was primarily used if the user anticipated a longer sequence of touch interactions. Other users always used *Tuck* to stow the pen.

B4. Grip vs. Pose. For each *grip* we observed a range of *poses*, often by wrist supination (turning the palm upward). Grasping motions with a pen therefore encompass the pattern of hand contact on the barrel, as well as the 3D orientation of the pen. We observed full palmar supination for *Tuck* and *Palm*, but only half-supination for the *Writing* grip.

B5. Extension Grips. We observed many grips where users extended one or more fingers while holding the pen to touch the screen, from both the *Tuck* and the *Palm* grips. However, not all extensions were observed from each grip; in particular, we did not observe any extension grips from the *Writing* grip, and the common *Tuck–Middle Extension Grip* did not occur with the *Palm* grip. (The 4-5 finger grips are logical, but our touch gestures required two fingers at most.)

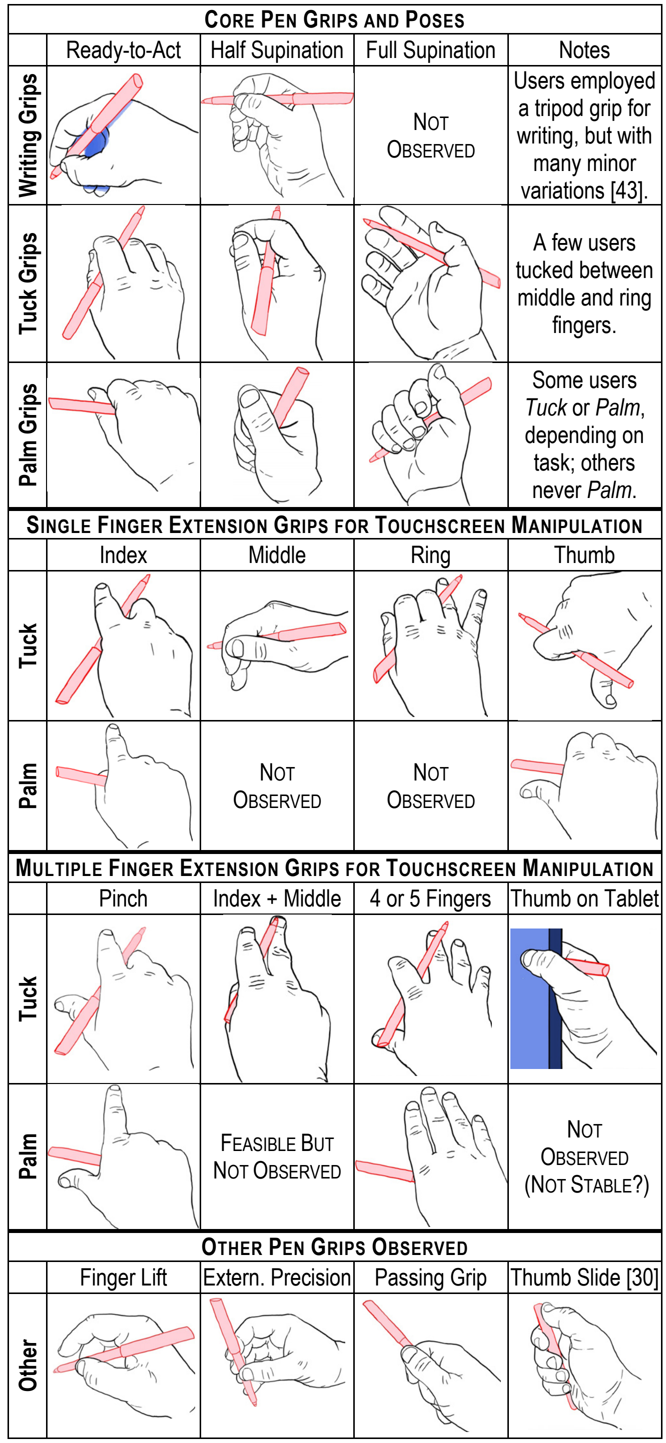

Figure 2. Taxonomy of pen + touch grips. See text for details.

B6. Variation in pen grips. Users exhibited many variations in the tripod grip for writing (see also [43]), leading to corresponding variations in users’ resulting *Tuck* and *Palm* grips, such as one user’s tendency to favor her ring finger for single-touch (Figure 2, *Tuck–Ring Finger Extension Grip*).

B7. Re-gripping. An implicit theme of observations B1-B6 is that users shift between a variety of grips. Furthermore, users often exhibited subtle adjustments to their grasp while writing and handling the pen. These naturally occurring *re-gripping behaviors* appeared to be motivated by comfort, fatigue, and functional considerations (such as reaching out a finger during an extension grip, as in B5).

B8. Consistency in grips. Each user consistently applied the same core pen grips in the same situations. Users also tended to maintain whatever grip required the least effort, until a perceived interaction barrier such as fatigue or inefficiency gave them an incentive to shift grips. Users switched tablet grips more often when sitting than standing, perhaps because there are few effective ways to hold a tablet while standing.

B9. Touchscreen avoidance behaviors. Users often adopted pen grips and hand postures, such as floating the palm above the screen while writing, or splaying out their fingers in a crab-like posture, to avoid incidental contact with the screen. Another form of touchscreen avoidance was perching the thumb along the outside rim of the tablet bezel, rather than letting it stray too close to the touchscreen when picking up the tablet with one hand. These unnatural and potentially fatiguing accomodations reflect the system's inability to distinguish the context of intentional vs. unintentional touch.

B10. Finger Lift for activating pen controls. Users activated the pen barrel button from the *Writing* grip, and then only with the index finger. Users held the pen still when tapping the button. The thumb is also potentially available for controls from the *Thumb Slide* grip, as noted by [30].

B11. External Precision Grip. Users employed this grip, with the pen held toward the fingertips [30] and perpendicular to the writing surface, for precise pointing at small targets, such as the web links in our web browsing task.

B12. Passing Grip. We also observed passing prehension [33] when participants passed the pen and tablet back to the experimenter. Users tended to hold the device securely, in more of a power grip, and extended it from their body while keeping it level, so that their intent was clear and so that the other person could grab it from the far side.

Having observed these natural behaviors, we now turn our attention to stylus + tablet hardware that can sense them. Note that it is not our intention to recognize all of these grips, or to leverage all of these behaviors, but rather to inform our approach and help shape many of our design decisions.

REALIZING GRIP+MOTION SENSING FOR PEN/TABLET

To support the range of context sensing techniques that we envisioned, we designed custom hardware to augment the stylus and tablet with inertial sensors and capacitive grip sensing, as well custom software to handle simultaneous pen + touch events from the touchscreen.

Pen Hardware Design

A flexible capacitive grid consisting of 7×30 sensing elements covers the entire barrel of the pen, which we wrap in heat-shrink tubing to protect the sensor and to provide a smooth and easy-to-grip cylindrical surface for sensing. The interior of the pen consists of a 3D-printed case that holds a miniature Wacom electromagnetic pen, a AAAA battery, and our custom circuitry. For inertial sensing we use the STMicroelectronics L3G4200D MEMS gyroscope as well as a LSM303DLHC accelerometer / magnetometer module. The capacitive grip sensor consists of a copper grid custom-printed on a flexible Kapton substrate, connected to a Cypress CY8CTMA463-60BUI touch controller. We stream data to a host using the 2.4GHz Nordic Semiconductor nRF24L01 transceiver operating at 2Mbps. An Atmel XMEGA 32A4U microcontroller runs our firmware.

We stream all inertial sensors off the pen at 130Hz, and the 7×30 capacitance map (Figure 3) at 30Hz. We designed our pen to be wireless; however, it proved difficult to change the battery without damaging the grip sensor, so we had to add a cable for power. The pen is 19cm long × 14mm diameter.

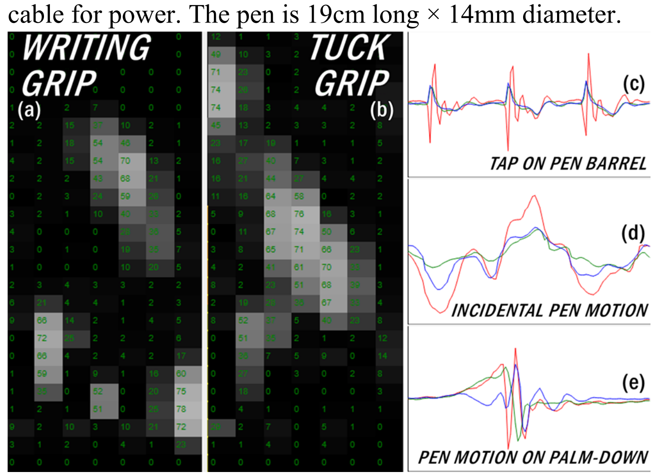

Figure 3. Example pen sensor data showing two different grips (a,b) and three example gyro signals (c,d,e). See text for details.

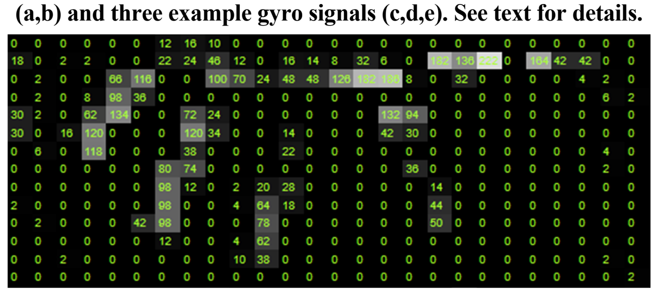

Figure 4. Example tablet case grip sensor data showing four fingers reaching in from the top to grasp the tablet.

Tablet Hardware Design

The tablet case (Figure 4) covers the entire back surface and sides of the tablet. The rigid case is constructed from printed circuit boards consisting of 44×26 capacitive sensing elements. There is a small insensitive area (in the middle of the case on the back side) where the integrated circuits are mounted. The case includes the same sensor components as the pen, except there are four touch controllers for different parts of the capacitive grid. We stream the tablet sensor data via USB, with the tablet's inertial sensors sampled at 100Hz and the tablet's 44×26 capacitance map sampled at 25Hz.

Simultaneous Pen and Touch from the Touchscreen

Because by default Windows deactivates touch when the pen comes in range (as a palm-rejection technique), we employ the Windows *Raw Input* API to capture multi-touch and pen events directly from the HID controller. Using this approach the *Samsung Series 7 Slate* we use for our prototype can report up to 8 touch contacts simultaneously with pen input.

Software Design

We time-stamp the pen / tablet sensor data on an external PC, compute grips, and then dispatch events to the tablet for presentation in the UI. Thus our implementaton consists of a distributed system with four cooperating components: the stylus, the tablet case, the tablet itself, and the external PC.

Inertial Sensor Fusion

We combine the accelerometer, gyro, and magnetometer using a direct cosine matrix algorithm ([31,37]). This produces stable yaw, pitch, and roll values in an east-north-up Earth coordinate frame, yielding the stylus orientation in a consistent reference frame relative to the tablet.

Stylus Grip Classification

At present we recognize four distinct grips: *Writing*, *Tuck*, *Palm*, and *No Grip* (for when the stylus is not held). Per observed behavior B4 (*Grip vs. Pose*), the grip recognition considers the pattern of hand contact (capacitive grip sensing) as well as the pose (orientation) of the stylus. We processed the incoming data to extract salient features and then trained a multi-class classifier to extract the stylus grips. We performed a multi-class classification of the stylus grip patterns by using a set of one-vs-all learners, where each learner was a Support Vector Machine (SVM) classifier. The result is a probability distribution over all four grips.

We selected features for grip classification that carefully considered the cylindrical symmetry of the stylus form-factor. Sensed grips should be agnostic to the roll angle of the pen; likewise, the size of the user's hand, or a grip higher or lower on the pen barrel, should not affect classification. Thus we compute a *normalized image* invariant with respect to grip height and roll angle. From the raw 7×30 capacitance map, we fit the non-zero capacitance values into a 7×10 normalized image. We shift the capacitance map in the y-dimension so that the first row of lit pixels corresponds to the bottom of the normalized image, and then scale the non-zero capacitance pixels to fit in the 10 rows of the normalized image. The features employed for grip classification therefore include the stylus yaw and pitch angles, the normalized grip image, as well as the normalized grip histogram in the x and y dimensions. We also include features for the number of lit (non-zero) pixels, and the pixel sum of all 7×30 capacitance values from the raw capacitance map.

Grip Training Dataset Collection

We recruited nine right-handed participants (4 female), all with prior exposure to pen/tablet use, to generate the training dataset. Per observation B7, grip classification must account for transitions between grips, grasp modification during extension grips (B5), and naturally occurring re-gripping behaviors. With this in mind, we designed our procedure to elicit such transitions. Our training data therefore incorporates common transitions between grips, as well as any re-gripping behaviors that users exhibited.

To collect all of the main grips for each user, we led users through a script illustrating specific grips and actions to perform in each grip. These included stowing the pen using touch (per observation B1) from both the Tuck and Palm grips (B2). We also included different sequences of tasks to capture the various common transitions between grips (B3). We led users through the full range of supination for each grip (B4) and included transitions between *Writing* and the single-finger and two-finger extension grips (B5), with articulation of direct-manipulation gestures such as tapping, dragging, and pinching. However, for writing tasks we did not specify any particlar tripod grip to use, but rather let users hold the pen naturally so that our dataset would capture cross-user variations in *Writing* grips (per B6). The data collection lasted approximately 15 minutes per user, with a total of 1200 samples for each user, per grip, yielding a total training dataset of 1200 × 3 grips × 9 users = 32400 samples.

Stylus Grip Recognition Accuracy

A10-fold cross-validation using the collected grip training dataset yielded an overall accuracy of 88% for the user-independent model. We also conducted a separate follow-up study with nine additional right-handed users, none of whom contributed to the training dataset. The task required users to write as well as perform specified touch gestures (tap, drag, and pinch) while holding the pen in the *Tuck* and *Palm* grips (i.e. it did not employ the same protocol used to collect the training dataset, to provide greater ecological validity). We assessed 8100 total samples (9 users × 3 grips × 300 samples at 250ms intervals) from the new users. This yielded user-independent grip recognition accuracy of 93% for the *Writing* grip, 93% for *Tuck*, and 77% for *Palm*. The relatively low recognition rate for the *Palm* grip appeared to stem from several users' tendency to hold the pen very lightly in this grip, resulting in a somewhat inconsistent pattern of contact sensed by the capacitive grip array. However, the system was still able to distinguish *Writing* vs. non-writing grips (i.e. Tuck *or* Palm) with 97% accuracy. Since the interaction techniques described below do not depend on any distinction between the *Tuck* vs. *Palm* grips, we proceeded with this user-independent grip model, which works well enough even without collecting training data for newly-encountered users.

GRIP + MOTION INTERACTION TECHNIQUES

Building on the behaviors we observed and the grip-recognition capability detailed above, our goal was to explore the design space of sensing techniques for tablet + stylus interaction. We explicitly sought to investigate techniques in the foreground of attention, such as new gestures made possible by additional degrees-of-freedom, as well as background sensing techniques that use context to automatically sense natural user behavior [6].

We pursued these goals in the context of a simple sketching application because we felt this would best showcase our ideas. Although sketching is a different scenario than the *writing side-by-side with reading the web* task used to generate observations B1-B12, we felt that the converse—

conducting behavioral observations on an already-existing sketching application—would not have yielded the same richness of observations. The behavioral study thus served as a point of departure, with insights that we now apply to our sketching application. It includes annotation, panning, and zooming capabilities representing a broad sample of popular pen-based applications (such as Microsoft OneNote, or Paper by FiftyThree [11]), and thereby provides various examples that illustrate the utility of our approach.

Pen Orientation Sensed Relative to the Tablet

Our inertial sensor fusion allows us to maintain a common reference frame relative to the tablet. We employ this tablet-relative orientation at the grip-recognition phase, as well as in the interpretation of the pen motion (such as for the airbrush tool, described later). It is important to emphasize here that we only can sense the *orientation* of the stylus relative to the tablet. Inertial sensing cannot reliabily determine the *(x,y)* translation or *z*-altitude of the pen without resort to some other absolute external reference.

PALM 'REJECTION' AS A PROBLEM OF CONTEXT

Sensing unintentional palm contact is a difficult problem because, at the onset of touch, there is often insufficient information to distinguish what type of touch is occuring. A palm *might* be recognizalbe as a touch with a large contact area, but such contacts always start small and may take a while to pass a certain size threshold. Also, some unintentional touches (such as contact produced by the knuckles) may never turn into a "large" contact. This strategy therefore necessitates delays (introduces lag) in processing touch events, and still may fail to detect many contacts [1,2].

Recently, Schwarz et al. demonstrated probabilistic palm rejection using the touch events after the palm makes contact with the screen [42]. Our approach is complementary because its cues (that the pen is held in a *Writing* grip or not) arrive before the hand touches down, or coincident with hand contact (in the form of the 'bump' motion signal).

To increase stability and avoid fatigue, users naturally rest their hand on the writing surface, but (per behavior B9), current tablet users are forced to adopt touchscreen avoidance behaviors. Simply sensing that the user is holding the pen is not sufficient to address this problem because people stow the pen while using touch (B1) and employ various extension grips to touch the screen (B5). Stylus orientation is also insufficient because each grip can be associated with a range of wrist supinations (B4) and because users hold the pen in a variety of ways (B6).

However, since unintentional touch primarily occurs incident to writing, sensing the *Writing* grip itself is a powerful cue, particularly because the user typically adopts a writing grip prior to resting his hand on the display. Hence, a highly conservative palm-rejection algorithm could simply reject any touch that occurs when the pen is held in the *Writing* grip. This, however, would preclude *intentional* touches made by the nonpreferred hand whenever the pen is held in the writing grip, which eliminantes many desirable scenarios (e.g. panning and zooming with the nonpreferred hand), as well as simultaneous pen + touch gestures [21].

With our sensors, we noticed that when the user plants his hand on the screen, it simultaneously induces a corresponding signal on the pen's motion sensors. (The tablet's motion sensors also pick up some of this motion, but at present we do not rely on this signal because it is damped somewhat by the greater mass of the tablet.)

The stylus motion exhibits a characteristic hard-contact profile similar to that seen with bump, whack, and thump gestures in other contexts ([9,16,23,41]). We look for a peak (using the second order finite difference [9] on the three combined axes of the accelerometer or gyro) that exceeds a minimum threshold within a 10-sample window. We also know exactly when to look for this signal because the palm produces a bump at the same time that the touchscreen detects the new contact. We can identify this peak within a maximum of 56ms after the touch-down event in the worst case (including all network delays).

This allows us to employ a resonable movement threshold for the bump signal, while also trivially rejecting other motion signals that do not occur coincident to a new touchscreen contact. However, our approach can and will miss subtle contacts, such as accidentally brushing the screen with a pinky, or if the user tries to fool the sensors by setting his hand down gently, for example. Nonetheless, we find our detection scheme works well for most normal pen interactions during writing.

For as long as the detected palm contact persists, we flag any new touches as a "palm" if they land within a 300 pixel radius. We found it helpful to provide feedback of the initial palm detection by playing a brief "radar circle" animation, centered on the palm-down location, that fades as it expands. This provides non-distracting feedback that confirms to the user that the palm contact was successfully detected. Without this feedback, the user may be left with a nagging uncertainty as to whether or not their palm has triggered an undesired action (such as calling up a menu, or leaving behind an ink trace) that is currently occluded by the hand.

Note that the *Paper* app by FiftyThree provides a *Pencil* hardware accessory [11] that activates a palm-blocking feature. However, the user must explicitly provide the context that the *Pencil* is in use by holding it on the screen to pair it. *Paper's* palm blocking further appears to only operate when the *Pencil* tip is actually in contact with the screen, which prevents it from sensing natural palm contact as the hand approaches. Hence, our context-sensing approach works very differently and can detect many types of palm contact that *Paper* and other related approaches [42] cannot.

Permitting Intentional Touch

Of course, the whole point of this approach is to permit simultaneous intentional touches, even when the palm is resting on the screen. With the above criteria, any new touch that occurs away from the palm (which is not accompanied by a bump on the pen) represents a true intentional touch. Our current implementation uses the first two additional touches that are not flagged as a palm contact to support the standard pinch-to-zoom gesture. Palm contact is ignored and does not interfere with pan/zoom, even if the palm moves.

However, because we still track the palm—rather than outright 'rejecting' it per se—our approach also can support techniques that use the palm location as an input, such as to help correctly orient menus [5] or to anchor objects [45].

DIFFERENTIATING PEN-IN-HAND vs. BARE HAND INPUT

The Magnifier Tool vs. Full Canvas Zoom

The functionality of our Magnifier Tool (Figure 1) was inspired by FiftyThree's *Paper* application [10]. *Paper* invokes a focus-plus-context magnifier (known as the "loupe") which is especially well suited to sketching tasks where the user wants to make a few detailed strokes without losing the overall context of the workspace. Unlike *Paper,* we support both the Magnifier *and* Full-Canvas Zooming by sensing how the user is interacting with the pen and tablet.

When the user stows the pen (via *Tuck* or *Palm*, per B2), we recognize this grip. If the user then brings two fingers into contact with the display, we check the stylus for a corresponding "bump" that occurs at the same time as the touch. When we see this combined stylus bump + two-finger touch, it brings up the Magnifier Tool. Users found the appearance of the tool in this context natural and appealing.

If we see a two-finger touch without any corresponding bump on the stylus, we instead infer that the user made the touch with their other (nonpreferred) hand, which is not holding the pen. This then triggers the standard two-finger pan and zoom interaction to allow full canvas zoom (rather than the focus-plus-context Magnifier Tool).

The Magnifier zooms only the region of the canvas under the circular tool. The magnifier interactively resizes itself according to the spread between the user's two fingers. The user may also touch down a finger on the border of the Magnifier to drag it to a new location. A single finger tap, or pen stroke, anywhere outside of the Magnifier dismisses it, leaving the canvas undisturbed at its original zoom level.

Note that, since we employ a minimum motion threshold to detect the bump signal, if the user touches their fingers down very lightly the stylus may not detect a motion signal sufficient to exceed this threshold. Nonetheless, our current thresholds are sufficient to detect the motions produced when users naturally bring their fingers to the screen with the pen stowed. Yet the way the sensing works thus affords an explicit work-around, by touching lightly, a noteworth point that crops up again in the informal evaluation reported near the end of this paper.

The Pen Controls

As another example, a single-finger tap while the pen is stowed brings up a small in-place palette containing the *Pen Controls*, allowing the user to change modes, or to modifiy the pen color and stroke thickness, without making a round-trip to a toolbar docked at the edge of the workspace. This example again takes advantage of the bump generated on the stylus when the user taps the touchscreen from an extension grip, using any single finger to make touchscreen contact (per B5). The tools appear next to the finger. The user may then interact with the radial menus using either pen or touch, as studies have consistently found that users expect pen and touch to be interchangable for UI controls [21,34].

FURTHER DIFFERENTIATING INPUTS BY PEN GRIP

The Drafting Tools

Our inspiration for the Drafting Tools arose from the observation that users often maintain the Writing grip between bursts of writing activity. For example, during pauses users often rotate the wrist away from the screen, to bring the pen into the *Writing–Half Supination* pose (B4). We reasoned, therefore, that the Writing grip itself represents an interesting context that might be explicitly supported by providing various Drafting Tools that take into account that the user is holding the pen in a ready-to-write posture.

The user calls up the drafting tools explicitly, by touching down a single finger of the nonpreferred hand (recognized by a single touch without a corresponding bump signal on the stylus). If the pen is held in the Writing grip, this brings up a small palette that currently offers two pen + touch tool modes, an *Airbrush* and a *Compass*, although we envision placing additional tools here (such as a straightedge, a french curve [25], pen + touch tape drawing [21], and so forth).

The current Drafting Tool is invoked as soon as the user touches down his finger; currently, the Airbrush is the initial default mode. The user can then tap on another tool (such as the Compass) to change modes. All drafting tools are implemented as spring-loaded modes; the mode is maintained only so long as the user holds down his finger.

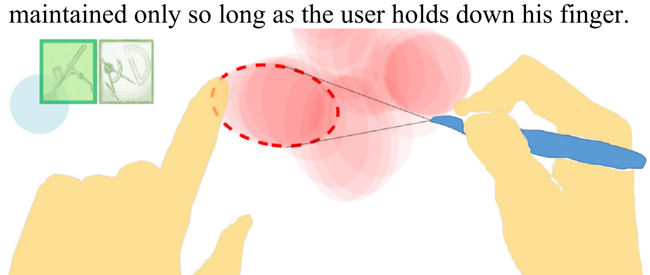

Figure 5. Airbrush tool. The nonpreferred hand indicates where to spray, while the pen orientation defines the conic section. The drafting tools palette (upper left) indicates the current mode.

Airbrush Tool

The Airbrush initially shows cursor feedback, as a gray dashed ellipse, of where the airbrush will spray if the user starts the paint flow. The user indicates where the airbrush tool should spray by the position of the (nonpreferred hand) finger. This is necessary because (as noted previously) inertial sensing cannot determine the absolute *(x,y,z)* location of the pen tip above the tablet, only a 3D orientation relative to the tablet. The relative sensing of the stylus orientation can be demonstrated by rotating the tablet, rather than the stylus, as shown in the accompanying video figure.

The user controls the spray ellipse by changing the azimuth and elevation angles of the pen with respect to the tablet surface. The user can hold the pen well above the tablet screen, making it easy to angle the pen as desired, unlike a previous exploration of an airbrush-like tool [47] which did not employ the stylus tilt angles, likely because it is difficult to reorient the pen while also keeping it within the limited ~15mm proximity sensing range of the tablet.

The user turns the paint flow on and off by tapping their index finger on the barrel of the pen (further detailed below). When the user activates the spray, the feedback changes to a bold red dashed ellipse to give clear feedback of the shape of the spray being produced. In our current prototype, we then "spray" highly transparent ellipses onto the canvas, which is sufficient for our proof-of-concept implementation.

Compass Tool

The Drafting Tools palette also includes a Compass, which supports a pen + touch mode where the pen is constrained to draw circular arcs centered about the current location of the finger (again of the nonpreferred hand). None of the extra degrees-of-freedom of the stylus are required by this mode.

Single-Tap Virtual Pen Barrel Button

Mechanical pen barrel buttons are convenient for menus and switching modes, but they force the user to hold the stylus in a particular grip [46] so that the finger can reach the button. They also can suffer from problems of accidental activation if the user mistakenly presses the button while writing.

To address these problems Song et al. [46] implement a virtual pen barrel button based on double-tap or finger slide motions, sensed through capacitive grip on the pen barrel. It is difficult to support single-tap interactions using capacitive grip alone, because many false single-touch contacts can be generated when the user shifts grips on the pen. Our system overcomes this limitation and successfully supports single-tap activation of a virtual barrel button by strategically combining all of the stylus sensing channels.

To identify candidate tap events, we look for a bump signal from the finger tap (*Figure 3c*) at the same time that a new touch contact appears on the capacitance image map. However, this cannot filter out all false positives produced by re-gripping the pen, because shifting grips can also produce bump signals coincident with new finger contacts.

To filter these out, we rely on our observation that users hold the pen still to maintain a stable tripod grasp when they lift the index finger to tap on the barrel (per observation B10). We therefore look at the ongoing accelerometer and gyro signals and compute a simple time-decaying motion signal similar to that employed by Hinckley et al. [19] to determine whether a mobile device is moving or held still. We then only accept candidate tap events that occur when the stylus is *not moving*, which effectively filters out any false contacts.

Note that the stylus must remain in a new moving (or not moving) state for at least 100ms. Otherwise, the pen barrel tap itself can trigger brief activation of a "moving" signal, which of course would thwart recognition of the barrel tap.

TABLET GRIP DETECTION TECHNIQUES

We use capacitive grip sensing on the back and sides of the tablet case to detect a number of additional contexts.

Thumb Menu and Handedness Detection

While we have so far focused on interactions involving the pen, there are many instances where the user picks up and holds a tablet with both hands, making the pen unavailable. We use grip to sense which hand the user is holding the tablet with. We then use this to summon a Thumb Menu at the appropriate side of the tablet, which allows the user to activate various buttons and menus directly with the thumb. If the user grasps the tablet with a second hand, we leave the Thumb Menu visible at the side where it first appeared.

If we observe the user grasping the pen while holding the tablet with one hand, we can immediately infer the user's handedness, although we do not yet use this to customize how menus appear, since our grip recognition is currently only trained for right-handed users.

Detecting Unintentional Thumb Contact

When the Thumb Menu first appears, it fades in over a 1.5 second interval; if the user lets go of the tablet it fades out after 350ms. The purpose of this animation feedback is to present the Thumb Menu in a tentative state, so that if the user's thumb strays onto the touchscreen while picking up the tablet, the menu can be ignored. We infer that a thumb is an unintentional touch if it occurs coincident with a new hand grip on the corresponding back portion of the tablet case. We then detect the thumb as an unintentional contact, and freeze the fade-in of the Thumb Menu if the unintentional thumb contact overlaps it. This feedback indicates to the user that the thumb contact has been recognized, but intercepted to prevent accidental activation of the menu. The user can then intentionally interact with the Thumb Menu, if desired, simply by lifting the thumb and bringing it back down on the menu. The fade-in animation then completes.

If the user does not place the thumb on the screen when picking up the tablet, the fade-in also serves as secondary cue that the Thumb Menu is fully ready for use. Since accidental activation mainly tends to occur when the user first grasps the tablet, after a few seconds elapse we assume that any hand contact with the screen was intentional. This therefore illustrates how our detection scheme blocks unintentional touch, while also allowing intention touches to get through, unlike the simple thumb-blocking heuristics on the iPad Mini, for example, which ignore any hand contact near the edge of the screen in certain applications.

Tablet Handoff: Passing the Tablet to Another User

We explored one final interaction, that of passing a tablet to another user as a way to offer an alternative, more physical semantic of sharing content with another user [18], much as observed during device micro-mobility [29,32]. Studies of passing prehension [33] and our observations (B12) indicate a characteristic sequence of motions. A user extends the object to offer it, while holding it level. He maintains his grip until the other person has firmly grasped the object. The person passing the object then lets go.

We employ our sensors to detect these states, using the tablet case grip sensing to determine when each user is grasping the device, and the tablet orientation to determine if the tablet is level. When these conditions are met, a special annotation layer peels over the tablet, as if a transparency or a sheet of vellum had been dropped over the display. The other user is then free to annotate the content, but not to make digital copies or navigate to other documents or files. This is a very different and much more limited form of sharing than the digital transfer of information supported by other cross-device information transfer techniques (e.g. [32]). We do not trigger Tablet Handoff when a single user holds the display up with two hands; during such interactions, users angle the tablet towards themselves, and thus it is not level.

Figure 6. Handing off the tablet drops down a layer for that user's annotations (shown by a page curl animation).

PRELIMINARY USER EVALUATION

We ran an informal study with 8 people (3 female, all right-handed, 23-51 years old, mean 37). All participants were office workers with previous experience on pen-operated devices with touch. To focus on preliminary user reactions, rather than how the techniques should reveal themselves, the experimenter demonstrated each technique, after which test users tried it themselves. Each session lasted 60-90 minutes.

Detecting Unintentional Palm Contact: The sensors detected many instances of palm contact that would not be picked up by approaches that require proximity of the pen tip to the display. However, there were still instances where the user brushed the screen or touched down lightly (e.g. with a pinky finger) that the sensors were not able to detect. Users were excited about the technique because many had suffered false touch inputs. One user commented he would easily get in the habit of landing with a "heavy" palm to ensure the sensors could always pick up motion. Users also appreciated the ability to pinch-to-zoom with the nonpreferred hand when the palm was already resting on the display.

Magnifier Tool: Users quickly understood the Magnifier, and were able to easily call it up. A couple of users performed an initial interaction with a light touch that the system could not detect, but they quickly adapted to making a more deliberate gesture that could be sensed. This suggests an opportunity to improve the detection by making it more sensitive to fine motions, but on the other hand users sometimes leveraged this range of expression to *intentionally* bypass the Magnifier. For example, one user stated he could employ this when using his tablet one-handed to selectively call up the Magnifier, or instead invoke pinch-to-zoom.

Pen Controls: Users were easily able to call up the pen controls by tapping with the pen tucked. However, users commented that depending on what they were doing with the pen, this could sometimes require them to change grips just to call up the tools. Several users proposed bringing up the pen controls using the virtual pen barrel button instead. We find this an interesting suggestion, but it is unclear *where* to bring up the menu if the pen is out of range when tapped.

Drafting Tools and the Airbrush: Users found the basic idea of calling up pen-specific drafting tools when they were holding the pen ready-to-write to be intuitive. Users found using the airbrush fun and exciting, but several commented that tapping the pen barrel to turn it on and off would disturb the location of the spray ellipse. Users also had a mixed reaction to the need to indicate where to spray by holding down a finger, because "your non-dominant had is doing the drawing, which is a little odd" as one participant commented.

Thumb Menu / Detecting Unintentional Thumb Contact: These capabilities resonated particularly well with our test users, and many felt strongly that it was natural to bring up a menu near to the thumb while also guarding against unintended thumb contact. Several users suggested improving the technique by using the location of the sensed grip to position the Thumb Menu closer to the thumb.

Overall Reactions: On the whole test users were excited by the system (commenting "it's like magic" and "this system is very cool, hopefully someday I can use it.") Users found the palm detection to be the most useful capability, as all had previously experienced this problem firsthand. The virtual barrel button was the lowest ranked feature, perhaps because its only demonstrated use was to turn on / off the Airbrush.

CONCLUSION: CHALLENGES, LIMITS, & FUTURE WORK

Although the present research makes considerable progress on many issues in sensing for stylus+tablet interaction, a number of challenges and limitations remain to be addressed:

- The thickness and length of our stylus prototype, as well as the presence of a tether for power, limit our ability to deploy it for truly mobile use and to observe a wider range of naturally-occurring behaviors with pen and tablet.
- Our current techniques for sensing pen-in-hand interactions require detecting some minimum force motion-signal threshold on the accelerometer and/or stylus. With more sophisticated signal-processing

techniques, or by combining the signal from the tablet's motion sensors as well, it may be possible to make the system more sensitive, but very soft touches likely represent an inherent challenge for this type of approach.
- Our observations and techniques need to be extended for left-hand users, to situational impairments such as mobility (i.e. walking) and one-handed interaction, and in general to a wide variety of other human accessibility concerns that our system does not yet address adequately.

Nonetheless, we have contributed a number of techniques that demonstrate how stylus and tablet sensing allows us to tailor pen+touch interaction to the context, such as by ameliorating unintentional palm contact while writing, yet still allowing full articulation of intentional touches by the nonpreferred hand. We can also combine these modalities to support contextually-appropriate tools, such as the Magnifier Tool, Drafting Tools (e.g. airbrush), and the Thumb Menu, all depending on how the user holds the tablet and pen.

A common theme in our techniques is the many ways that motion sensing, grip sensing, and multi-touch from the touchscreen can be combined to reinforce one another, both to avoid false positives as well as to help extract specific signals that would otherwise be difficult to isolate from a single sensing modality. Collectively, these techniques illustrate some of the compelling ways that grip and motion sensing afford contextual awareness in pen computing, and thereby open up the potential for a rich new space of expression for tablets using natural pen + touch interaction.

REFERENCES

1. Annett, M., et al. The Pen is Mightier: Understanding Stylus Behaviour While Inking on Tablets. *Graphics Interface (GI 2014)*.
2. Annett, M., et al. Is it Intended or Unintended? Palm Rejection during Direct Pen Interaction*.* (Currently submitted for review), 2014.
3. Becchio, C., et al. Grasping intentions: from thought experiments to emprical evidence*.* Frontiers in Human Neuroscience, 2012. **6**(117).
4. Bi, X., et al. An Exploration of Pen Rolling. *UIST'08*.
5. Brandl, P., et al. Occlusion-aware menu design for digital tabletops. *CHI'09 Extended Abstracts*.
6. Buxton, W. Integrating the Periphery and Context: A New Taxonomy of Telematics. *Proc. Graphics Interface '95*.
7. Cheng, L.-P., et al. iRotate grasp: automatic screen rotation based on grasp of mobile devices. *UIST Adjunct Proc. '12*.
8. Cheng, L.-P., et al. iGrasp: grasp-based adaptive keyboard for mobile devices. *CHI '13 Extended Abstracts*.
9. Chun Yat Li, F., et al. The 1Line Keyboard: A QWERTY Layout in a Single Line. *UIST'11*.
10. FiftyThree, I. *A closer look at zoom*. http://tinyurl.com/mxqdshz.
11. FiftyThree, I. *Pencil (for Paper)*. http://www.fiftythree.com/pencil.
12. Goel, M., et al. GripSense: Using Built-In Sensors to Detect Hand Posture and Pressure on Commodity Mobile Phones. *UIST'12*.
13. Guiard, Y., Millerat, F., Writing Postures in Left-Handers: Inverters are Hand-Crossers*.* Neuropsychologia, 1984. **22**(5): p. 535-538.
14. Harrison, B., et al. Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. *CHI'98*.
15. Hasan, K., et al. A-coord input: coordinating auxiliary input streams for augmenting contextual pen-based interactions. *CHI '12*.
16. Hinckley, K. Synchronous Gestures for Multiple Users and Computers. *UIST'03*.
17. Hinckley, K., et al. Motion and Context Sensing Techniques for Pen Computing. *Proc. Graphics Interface 2013 (GI'13)*.
18. Hinckley, K., et al. Codex: A Dual-Screen Tablet Computer. *CHI'09*.
19. Hinckley, K., Pierce, J., Sinclair, M., Horvitz, E. Sensing techniques for mobile interaction. *UIST 2000*.
20. Hinckley, K., Song, H. Sensor Synaesthesia: Touch in Motion, and Motion in Touch. *CHI 2011*.
21. Hinckley, K., et al. Pen + Touch = New Tools. *UIST 2010*.
22. Holmquist, L., et al. Smart-Its Friends: A Technique for Users to Easily Establish Connections between Smart Artefacts. *Ubicomp'01*.
23. Hudson, S., et al. Whack Gestures: Inexact and Inattentive Interaction with Mobile Devices. *TEI 2010*.
24. Kim, K.-E., et al. Hand Grip Pattern Recognition for Mobile User Interfaces. *Proceedings of AAAI/IAAI-2006: Innovative Appl.*
25. Kurtenbach, G., et al. The design of a GUI paradigm based on tablets, two-hands, and transparency. *CHI'97*.
26. Lee, D., et al. PhantomPen: virtualization of pen head for digital drawing free from pen occlusion & visual parallax. *UIST'12*.
27. Lester, J., et al. Are You With Me? Using Accelerometers to Determine if Two Devices are Carried. *Pervasive'04*.
28. Liu, S., Guimbretiere, F. FlexAura: A Flexible Near-Surface Range Sensor. *UIST '12*.
29. Luff, P., Heath, C. Mobility in collaboration. *CSCW '98*.
30. Mackenzie, C., Iberall, T., *The Grasping Hand*. Advances in Psychology 104. 1994, Amsterdam: North Holland.
31. Mahoney, R., Nonlinear Complementary Filters on the Special Orthogonal Group. IEEE Trans. on Automatic Control, 2008. **53**(5).
32. Marquardt, N., et al. Cross-Device Interaction via Micro-mobility and F-formations. *UIST '12*. 2012. Cambridge, MA.
33. Mason, A., MacKenzie, C., Grip forces when passing an object to a partner*.* Exp. Brain Res, 2005(163): p. 173-187.
34. Matulic, F., Norrie, M. Supporting active reading on pen and touch-operated tabletops. *Proc. Int'l Working Conf. on Advanced Visual Interfaces (AVI'12)*. 2012. Capri Island, Italy.
35. Moleskine Journal (iPad app). 2014. http://tinyurl.com/kqhtomz.
36. Penultimate by Evernote (iPad app). 2014. http://tinyurl.com/kx82c8a.
37. Premerlani, W., Bizard, P. *Direction Cosine Matrix IMU: Theory*. 2009. http://gentlenav.googlecode.com/files/DCMDraft2.pdf.
38. Rofouei, M., et al. Your Phone or Mine? Fusing Body, Touch and Device Sensing for MultiUser Device-Display Interaction. *CHI 2012*.
39. Roudaut, A., et al. TimeTilt: Using Sensor-Based Gestures to Travel through Multiple Applications on a Mobile Device. *Interact '09*.
40. Schmidt, A., et al. Advanced interaction in context. *Handheld and Ubiquitous Computing (HUC'99)*.
41. Schmidt, D., et al. PhoneTouch: A Technique for Direct Phone Interaction on Surfaces. *UIST'10*.
42. Schwarz, J., et al. Probabilistic Palm Rejection Using Spatiotemporal Touch Features and Iterative Classification. *CHI'14*.
43. Selin, A., *Pencil Grip: A descriptive Model and Four Empirical Studies*. 2003: Akademi University Press.
44. Sellen, A., Shaw., K. How Knowledge Workers Use the Web. *CHI'02*.
45. Siio, I., Tsujita, H. Mobile interaction using paperweight metaphor. *UIST '06*.
46. Song, H., et al. Grips and Gestures on a Multi-Touch Pen. *CHI'11*.
47. Sun, M., et al. Enhancing Naturalness of Pen-and-Tablet Drawing through Context Sensing. *Interactive Tabletops and Surfaces (ITS '11)*.
48. Suzuki, Y., et al. Stylus Enhancement to Enrich Interaction with Computers. *HCII 2007, Part II*. 2007: Springer-Verlag.
49. Taylor, B., Bove Jr., V. Graspables: Grasp-Recognition as a User Interface. *CHI'09*.
50. Tian, F., et al. Tilt Menu: Using the 3D Orientation Information of Pen Devices to Extend the Selection Capability. *CHI'08*.
51. Vogel, D., Casiez, G. Conte: Multimodal Input Inspired by an Artist's Crayon. *UIST 2011*.
52. Wagner, J., et al. BiTouch and BiPad: Designing Bimanual Interaction for Hand-held Tablets. *CHI'12*.
53. Wimmer, R., Boring, S. HandSense - Discriminating Different Ways of Grasping and Holding a Tangible User Interface. *TEI '09*.
54. Xin, Y., et al. Natural Use Profiles for the Pen: An Empirical Exploration of Pressure, Tilt, and Azimuth. *CHI'12*.
55. Yoon, D., Chen, N., Guimbretiere, F. TextTearing: Expanding Whitespace for Digital Ink Annotation. *UIST'13*.