Turbulent Touch: Touchscreen Input for Cockpit Displays

Andy Cockburn[1], Carl Gutwin[2], Philippe Palanque[3], Yannick Deleris[4], Catherine Trask[2], Ashley Coveney[2], Marcus Yung[2], Karon MacLean[5]

[1] University of Canterbury, Christchurch, New Zealand, andy@cosc.canterbury.ac.nz
[2] University of Saskatchewan, Saskatoon, Canada, gutwin@cs.usask.ca
[3] Université Paul Sabatier, Toulouse 3, France, palanque@irit.fr
[4] Airbus Operations, Toulouse, France, yannick.deleris@airbus.com
[5] University of British Columbia, Vancouver, Canada, maclean@cs.ubc.ca

ABSTRACT

Touchscreen input in commercial aircraft cockpits offers potential advantages, including ease of use, modifiability, and reduced weight. However, tolerance to turbulence is a challenge for their deployment. To better understand the impact of turbulence on cockpit input methods we conducted a comparative study of user performance with three input methods – touch, trackball (as currently used in commercial aircraft), and a touchscreen stencil overlay designed to assist finger stabilization. These input methods were compared across a variety of interactive tasks and at three levels of simulated turbulence (none, low, and high). Results showed that performance degrades and subjective workload increases as vibration increases. Touch-based interaction was faster than the trackball when precision requirements were low (at all vibrations), but it was slower and less accurate for more precise pointing, particularly at high vibrations. The stencil did not improve touch selection times, although it did reduce errors on small targets at high vibrations, but only when finger lift-off errors had been eliminated by a timeout. Our work provides new information on the types of tasks affected by turbulence and the input mechanisms that perform best under different levels of vibration.

Author Keywords

Touch interaction; turbulence; aviation.

ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION

Commercial aircraft cockpits are replete with physical controls, including many forms of switches, knobs, levers, dials, keypads and wheels. While computer-based flight instrument displays are becoming increasingly prevalent (i.e., the ‘glass cockpit’), these displays are almost exclusively used for data output, and user input to displayed objects is dependent on a separate indirect device. When a cursor is incorporated in the display, a trackball is typically used for item selection.

The reliance on physical controls is influenced by several factors, including the historical development of cockpit environments, pilot expectations, and requirements for safety, redundancy and adherence to standards imposed by regulatory authorities. For example, Boeing and Airbus conform to the ARINC 661 standard [1] that stipulates behavioral requirements for GUI components in a Cockpit Display System (CDS). The standard permits only a limited set of widgets, and it is slow to evolve – the explicit account for cockpit touch input first appeared in the 2016 update [2].

Touch interaction in commercial cockpits offers potential advantages to pilots, airlines, and aircraft manufacturers. Pilots may gain from familiar, expressive, and direct means for interaction in certain tasks. Airlines and manufacturers could gain from reduced hardware installation complexities. Replacing hard-wired physical controls with touchscreens could therefore ease development, facilitate upgrades, reduce weight, and improve pilot interaction.

However, turbulence is a challenge for cockpit touchscreen deployment. There are risks that the pilot’s ability to interact with the touchscreen may be substantially impaired or eliminated during periods of heavy cockpit vibration. Previous studies have shown that vibration can be a factor in touch input, but there are few studies that look at different types of interactive touch tasks or that consider how the problem of turbulence might be reduced.

We therefore examined users’ ability to interact with various types of interactive objects at different vibration levels. We used a motion platform to expose participants and a large touchscreen to three levels of simulated turbulence – none, low, and high. All tasks were completed using three different means for input – a trackball (used in many commercial aircraft), a 21.5 inch touchscreen, and the same touchscreen augmented with a guiding stencil overlay. The stencil

CHI 2017, May 06-11, 2017, Denver, CO, USA

DOI: http://dx.doi.org/10.1145/3025453.3025584

overlay consisted of a 3mm-thick transparent Lexan sheet that entirely covered the touchscreen (see Figure 1). Holes were cut through the sheet to permit interaction with underlying widgets. The sheet's 3mm thickness eliminated capacitive sensing of the finger outside the cut-out regions. The stencil overlay provided two potential benefits for turbulent touchscreen interaction. First, it might offer stabilization benefits because users can place their fingers or hand-edge on the stencil, without contact registration, and subsequently move their finger to the target (by sliding over the stencil and 'popping' into a hole). Second, once within a hole, users can further stabilize movement by pushing against the stencil's edge.

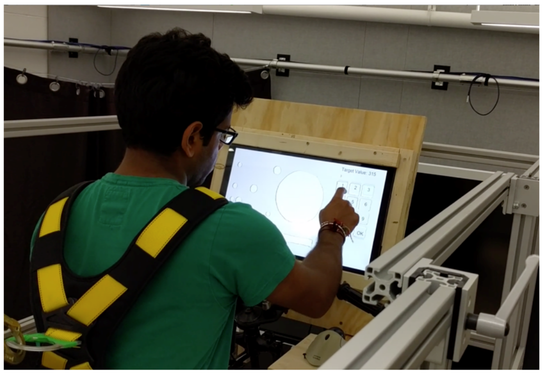


Figure 1: Experimental set up. Participants sat on a motion platform and made selections using a touchscreen or trackball. A transparent stencil (shown) was overlaid on the touchscreen.

Results include the following findings: touch interactions were faster and preferred at low vibrations; the stencil assisted touch selection of small targets at high vibrations; the stencil increased accidental lift-off errors, but this can be accommodated by using a short timeout between touch selections; drag-based selections were highly inaccurate during vibration with touch and trackball; and multi-touch pan-and-zoom interactions were much faster than the equivalent trackball method, even at high vibrations.

We make three main contributions: we demonstrate that different tasks have very different tolerance to turbulence; we provide empirical data on the effectiveness of touch and trackball input at high levels of vibration; and we test the value of a novel stabilizing stencil overlay for touchscreens. Overall, we provide new information for designers who must accommodate high vibration in touch-input.

BACKGROUND

Aviation Cockpits and Controls

The lifespan of a passenger aircraft is typically 20-30 years, operating between 35,000 and 110,000 cycles dependent on longhaul or shorthaul use. During its lifetime, much of an aircraft's componentry will be replaced and updated, including power-units, jet engines, and parts of the airframe.

Updates to cockpit controls are also desirable, particularly when new technologies become available. However, cockpit updates can be prohibitively expensive because controls predominantly rely on mechanical switches, dials, and levers with wiring redundancy (in case of failure on one channel). The reliance on hardware stems from the aviation regulatory environment, including conformance with standards.

While physical controls retain many advantages for certain flight-critical tasks, there is substantial industry interest in moving functions to cockpit display systems. Examples include the Lockheed Martin F-35 Lightning II stealth fighter, which has a large touchscreen display [9], Garmin International's patent for dual touch/cursor control of a CDS [22], Rockwell Collin's cockpit touchscreens [13] for light jets and turboprops, and the Thales Group "Avionics 2020" vision for the cockpit of the future [42]. See Kaminani [20] and Hamon [12] for reviews.

In empirically examining touchscreen interaction in the cockpit, Stanton et al. [33] analyzed the effectiveness of four different input devices (trackball, rotary controller, touch pad and touchscreen) for menu navigation in non-turbulent environments. They concluded that different devices have different strengths, and that the touchscreen had the highest number of 'best' scores. Barbé et al. [4] and Huseyin et al. [17] examined ergonomic aspects of touchscreen positioning in airplanes and helicopters. Wang et al. [37] recently examined the influence of the size and shape (square versus rectangular) of touchscreen targets. Targeting time results were consistent with Fitts' law, errors were higher with rectangular targets, and participants preferred larger targets to small ones. In another cockpit-directed targeting study, Lewis et al. [24] compared left-handed targeting performance when using a touchscreen and trackpad (left-handed performance was studied based on the assumption that the right hand would be on a control stick, which is not the case for the captain in a passenger aircraft). The main finding was that participants were faster when using the touchscreen than they were with the trackpad. Neither of these studies examined the influence of turbulence.

In addressing issues arising from turbulence, at least two patent applications disclose methods to facilitate the use of touchscreens in the cockpit. Kolbe [21] describes a method for determining whether touchscreen events are invalid following detection of a 'movement indicative of turbulence' by analyzing the characteristics of the contact data. The method of Komer et al. [22] instead assures that an alternative cursor-based indirect method of input (such as a trackball) is available to supplement touchscreen control.

In an early empirical study of cockpit touch interaction, Bauersfeld [5] compared two different touchscreen activation modes for confirming selection actions – confirm on contact versus confirm on lift-off (i.e., when the finger is released from the surface). These modalities were compared at two different levels of simulated turbulence, using a NASA-Ames flight simulator. Results confirmed the higher accuracy of lift-off observed in prior findings in non-turbulent environments [30]. More recently, Hourlier et al. [16] examined the perceived realism of cockpit turbulence simulations. Participants used an 11 point rating scale (from 0 for "not realistic at all" to 10 for "extremely realistic") to assess the perceived realism of turbulence that was simulated using a hexapod robot similar to that used in our experiment. Mean ratings ranged from 6 for the simulation of light turbulence to ~9 for the simulation of severe turbulence.

Huseyin et al. [18] recently reported results of a Fitts' Law study examining the influence of simulated constant G-forces on touchscreen target acquisition. Participants wore a weighted bag on their wrist while acquiring 55px (15mm) and 75px (20mm) targets. Results showed that acquisition times increased with wrist weight. Finally, Dodd et al. [8] examined the impact of moderate levels of simulated flight turbulence on touchscreen virtual-keypad input, with buttons of different sizes and separations. Data entry times, error rates, fatigue, and perceived workload were all higher at moderate turbulence than in a no-turbulence condition.

Touchscreen interaction and input stabilization

Abundant research has examined the efficiency and accuracy of different methods for interacting with various forms of touchscreens. Early studies compared user performance (time and error rate) attained when different methods are used to terminate touch selections, such as first-contact, slide-over, and lift-off [26, 30, 31]. Dragging actions with different input devices have also been closely examined (e.g., [7, 10]). In general, studies suggest that direct finger pointing is fast but relatively imprecise unless augmented with a cursor-like indicator (e.g., [36]), that stylus input is fast but can be inaccurate, and that indirect pointing methods can be highly accurate. Zhai et al. [43] provide a comprehensive review of work on gestural touch interaction.

Several studies have examined methods for enabling interaction by users with motor impairments, such as uncontrolled tremor. Barrier Pointing [11] proposed stylus-based methods for target acquisition using bezel edges and corners of a mobile device to facilitate input stabilization. Similarly, EdgeWrite [40] used the ridge edges of a small trackpad region to improve text-entry movements for users with limited motor control. Related studies have examined trackball and joystick input for similar objectives [38, 39].

The work most closely related to our current study concerns computer input in motion environments. In an early study of ship motion, McLeod et al. [27] showed that joystick input was negatively affected by motion across a variety of tasks. Hill and Tauson [15] showed that soldiers' computer input in vehicles (such as an armored personnel carrier) is negatively influenced by vehicle vibration. Similar findings have been shown for mouse input on trains [29].

Yau et al. [41] examined user performance in abstract target selection activities using three types of trackball and a touchscreen in five different settings based on ship movement (static, 'heave', 'roll', 'pitch' and 'random', all at 0.3Hz, with a maximum vertical displacement of ±100mm). Results indicated the superiority of the mouse in vibration environments, as well as showing that touchscreen errors increase with vibration. Similarly, Lin et al. [25] based their studies on ship motion, comparing the influence of low-frequency vibrations on touchscreen, mouse, and trackball input using a Fitts' Law target selection methodology. Their maximum vibration condition used root mean square (RMS) accelerations of 0.34m/s^2, corresponding to the lower end of 'a little uncomfortable' on the ISO vibration discomfort index for low frequency vibration (ISO 2631-1). Their findings indicated that the touchscreen allowed fast but inaccurate input and that the trackball was slow but accurate.

Neither of the ship studies examined the higher frequency vibrations and much higher accelerations incurred during flight. While the study of Dodd et al. [8] did examine touch input during simulated flight turbulence, they did not examine other types of input activity beyond keypad entry (such as dragging or multitouch pan/zoom operations), nor did they compare touch with other input methods. None of the studies examines methods for remediating touch input problems during turbulence.

TOUCHSCREEN STENCIL OVERLAYS

The success of edge-based touch stabilization for users with motor impairments suggests that related methods might improve touch input in turbulent environments. In particular, there are opportunities to assist stabilization on interface widgets by placing a transparent stencil overlay on top of the touchscreen. The idea of using a stencil or template to allow users to feel touchable parts of a touch panel was first described by Buxton et al. [6].

Like Buxton's template, our stencil contains holes that match the geometry of underlying touchscreen widgets, offering two potential interaction benefits. First, users may be able to stabilize their hand and fingers during target approach by placing fingers or palm on top of the overlay, without triggering touch registrations on the touchscreen. Users could then slide their finger into a hole to acquire a widget. Second, once one or more fingers rest within a stencil hole, users could press against the stencil edge to stabilize their control over dragging actions (e.g., when operating a slider).

However, it is possible that the stencil may slow certain interactions. For example, Avrahami [3] showed that touch target acquisition times were slower by up to ~90 ms when targets were located close to the edge of a displayed border.

Stencil Thickness, Material, and Friction

Multitouch touchscreens predominantly use projected capacitive sensing to detect charged objects on, or in close proximity to, the surface. Objects are not sensed beyond a few millimeters from the display – a 3mm-thick sheet of Lexan polycarbonate prevented touches except where holes were cut. Lexan has excellent optical clarity, is easily worked, has good abrasion resistance, and is widely used in the aerospace industry for aircraft window dust covers.

It is likely that there is a friction sweet-spot that best affords contact stability without compromising control due to 'stickslip' effects. Robinson et al. [32] examined surface friction in cockpit touchscreen environments, and Levesque et al. [23] examined methods for programmatically varying touchscreen contact friction.

Hole Size and Edge Profile
The shape of the edge of the stencil holes, and the size of the holes with respect to the underlying widget, will influence several aspects of interaction, including comfort, accuracy, friction, and stability provided by the edge. We considered various edge profiles, including those shown in Figure 2.

The bevel edge shown in Figure 2a should be relatively comfortable, but it is likely to provide low levels of edge stability. The square edge shown in Figure 2b should be more stable, less comfortable, and may create a small gap between the stencil edge, finger, and contact surface. The ∑ profile edge shown in Figure 2c, is likely to be yet more stable, but uncomfortable, and with a small possible gap between stencil base and finger at the contact surface. The blade edge of Figure 2d is also likely to be stable and uncomfortable, with a potentially large gap from stencil base to finger contact.

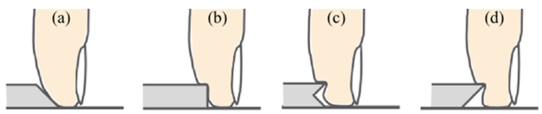


Figure 2: Candidate stencil edge profiles, from most (left) to least (right) comfortable (and least to most stable).

Our experiments used the square edge shown in Figure 2b, representing a compromise between comfort, stability, and potential stencil-finger surface gap. To account for the anticipated gap, we made the stencil holes approximately 0.5 mm wider on each side than the target widgets (i.e., 1 mm diameter larger for circular targets).

Supporting Mode Transitions
Cockpit display systems often support modes that present different interfaces, such as control panel (ECAM) in one mode and navigation information in another. Various techniques could be used to reconfigure the overlay when mode changes occur. If stencils were fabricated from a flexible material, the overlays could be stored on a roller housed beneath the display. When a mode change occurred, actuators could roll the required stencil into place. Alternatively, actuators might pop-up a new stencil from a 'toast-rack' of candidates when needed, leaving the flight crew to snap the stencil into a housing.

Input Redundancy
While stencil overlays may assist touch input during turbulence, the need for safety critical operation requires high redundancy in the cockpit. Therefore, redundant methods of input would need to be supported, such as the availability of a trackball and cursor to augment or replace a touchscreen in the event of hardware failure. This need is directly addressed by a patent to Komer et al. [22].

EVALUATING INPUT METHODS IN TURBULENCE
We conducted an experiment to better understand the influence that simulated turbulence has on interaction. Participants completed a series of five interactive task types using each of three different input methods (trackball, touchscreen, stencil-touchscreen) at each of three different vibration levels (none, low, and high). Trackball performance serves as a baseline comparator, representing current industry practice for cockpit interaction with CDSs.

Turbulence was simulated using a motion platform. Participants wore a safety harness connected to the ceiling and sat on a seat mounted on the platform. The harness did not encumber arm movement, and was present only to prevent falling. The display, also mounted on the motion platform, was placed approximately 45cm in front of the participant's chest, with the top edge of the display ~25cm below the user's eye-line. This arrangement facilitates easy touch interaction without interfering with the pilot's view from the cockpit windows [4].

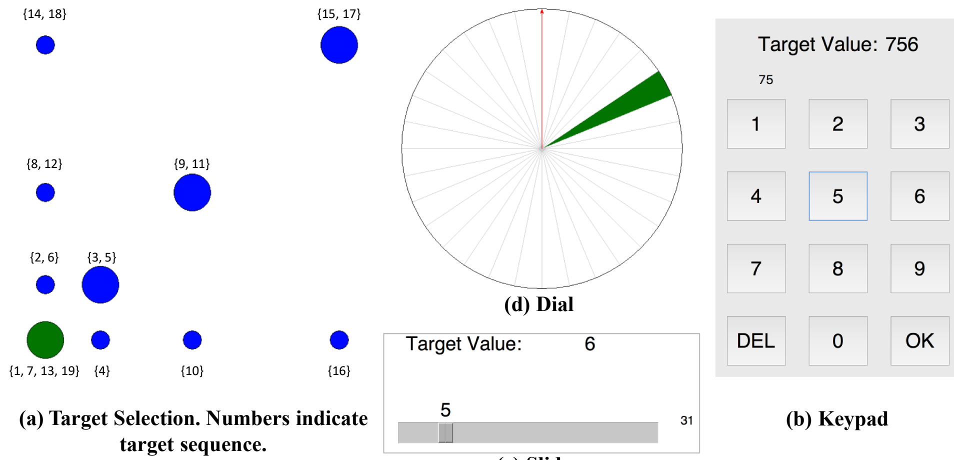
The objectives of the study were to compare and examine users' performance (task accuracy and error rates) with the three devices for various activities during vibration.

Task Types
Five different types of experimental tasks were used with each input device at each vibration level. These five task-types were as follows.

Target selection – participants tapped/clicked on targets as quickly and accurately as possible. There were three movement amplitudes (96, 256 and 512 px) and two target sizes (32 and 64 px). Figure 3a shows the arrangement of targets, with the annotated braces showing the target sequence – successive movement directions were N, E, S, N, W, S for each of small, medium and long distances. Each target was highlighted green, reverting to blue when successfully tapped. Each successful tap advanced to the next target, highlighting it green. Any error caused the background to turn red for 500 ms.

Keypad tasks involved entering and confirming each of four different three-digit numerical values. Each target value was cued at the top of the window (Figure 3b). Once the three digits were entered, the participant pressed "OK" to confirm the value. Each button in the keypad was 104 px wide. Keypad interaction is a common activity for pilots, such as entering radio frequencies or waypoint coordinates.

Feedback for each keypress was shown immediately below the target value. Incorrect entries (on pressing 'OK') resulted in the cue and entered numbers highlighting red for 500 ms; users then had to correct their error (using the 'Del' key to delete erroneous digits) and correct their entry. Correct values were highlighted green for 500 ms, then the next target value was shown. Each three-digit target value comprised two digits from the edge of the keypad (1, 2, 3, 4, 6, 7, 9) and the digit '5' (centre of the keypad).



(a) Target Selection. Numbers indicate target sequence. **(d) Dial** **(c) Slider** **(b) Keypad**

Figure 3: Target Selection (left), Keypad (right), Slider (bottom) and Dial (top) tasks shown in their approximately location within the experimental display. The Target Selection figure is augmented with braces showing the target sequence.

Slider tasks involved setting a slider to a target value (Figure 3c). This task emulates a CDS version of setting a physical lever to a target value. The target value was shown above the slider, and a slider permitted selecting values between 0 and a maximum value shown at the right of the slider (see Figure 3c). Lifting from the display, or releasing the trackball button, completed the selection. Correct selections were confirmed by green highlighting for 500 ms, then the next trial began automatically. Incorrect selections caused red highlighting for 500 ms, and the trial continued.

The slider length was 512 px. The number of candidate items within the slider varied across trials to control the precision required. The full set of trials comprised three repetitions at each of the following resolutions: *2, 8, 32, 128* candidate items in the slider. For each trial, the target value was randomly selected between 1 and the maximum value; and the slider was initially set to 0 for each trial.

Dial tasks involved setting a dial needle to a highlighted target in the dial (see Figure 3d). The needle could be set to a location either by tapping/clicking in the target area (suitable when the target segment was large) or by dragging the needle with the finger/cursor. Software checked the correspondence between the needle and intended target when the finger was lifted from the display (or trackball button released). If correct, the dial flashed green for 500 ms and the next target segment was highlighted, with the needle initially at due north. If incorrect, the dial flashed red for 500 ms and the trial continued. The resolution of task precision was controlled across trials by varying the number of candidate items in the dial. The full set of trials consisted of two repetitions at each of the following resolutions: *4, 8, 32, 128.*

Pan-zoom tasks required moving a yellow highlighted target circle into the view and manipulating scale until the circle was sufficiently large to become selectable (it turned green). The task emulates the activity of selecting destinations or waypoints on a map. The display contained two circles labelled 'left' (on the left) and 'right' (horizontally right of the left target). The first target (left) was displayed at a scale factor of 0.5. Touchscreen and stencil users zoomed into the target through multitouch manipulation of the surface ('pinch to zoom' or its bimanual equivalent). Single finger contacts panned the display at its current scale factor. Trackball users panned by dragging the ball with the button depressed, and zoomed by rotating the scrollwheel, with the cursor location serving as the scale orientation point.

The target turned green and became selectable at scale factor 1.3 or greater. Having selected the left target by tapping/clicking on it, it turned grey and the right-hand target was highlighted. Users then zoomed out to view the next target, and combined pan and zoom actions to bring it to a selectable scale factor. Software implemented maximum and minimum scale factors of 3.0 and 0.04. Software prevented panning to illogical regions, such as moving the left target beyond the right display edge. The full set of trials consisted of four target selections, beginning with an initial selection of 'Left' (data discarded, as unlike other trials it began with the target in view).

Procedure

On arrival, participants were briefly shown the motion platform, harness, and apparatus. They were informed that the objective of the study was to compare the effectiveness of different means for input in turbulent environments. Having put on the harness, they sat on the platform seat, and the ceiling-mounted securing cables were adjusted.

For familiarization, participants completed a training set of trials consisting of all input methods and task type. The motion platform was turned off during this familiarization.

Participants then advanced to the main experimental tasks, which progressed according to the following algorithm:

for each vibration ∈ {none, low, high}:
  for each input device ∈ {trackball, touch, stencil}
    for each task ∈ {targets, keypad, slider, dial, panzoom}
      complete task trials
  complete subjective experience questionnaire

Participants could orally request that any trial be abandoned. The experimenter then pressed a control key to advance to the next task, and the abandonment was logged.

Order of vibration level and input device was counter-balanced. Tasks types were always completed in the order *targets, keypad, slider, dial, panzoom.*

Having completed all tasks with all three input devices at one vibration level, participants completed a subjective experience questionnaire, recording NASA TLX measures for physical workload and frustration, subjective preferences, and other comments. While the participant completed these worksheets, the experimenter loaded the next vibration profile into the motion platform.

Apparatus
Software for the experiment was written in Python 3.4, using the kivy package for multitouch interaction in the pan-zoom tasks. Software ran on a Dell S2240T 21.5″ monitor running at 1920 × 1080 pixel resolution (4.03 px/mm). The monitor and participant's seat were mounted on a Mikrolar R-3000 hexapod motion platform. The experimenter sat alongside the motion platform, configuring changes to motion profile and input device when instructed by the software.

To avoid continually switching between different stencils for the five task-types, a single stencil was used for all task types. Each task type therefore inhabited a different region of the display – target tapping was top-left; keypad top-right; dial top-centre; slider bottom-centre (the transparent stencil overlay is visible in Figure 1). During pan-zoom tasks with the stencil, participants were instructed to use the dial hole for controlling interaction.

The motion pattern sent to the hexapod platform was designed to induce non-periodic vertical displacements with a mean motion frequency of 3.1 Hz (max 5 Hz). Movement amplitudes were configured to produce weighted RMS accelerations in the 'very uncomfortable' range (ISO2631-1, $1.25 < a_T < 2.5$ m/s^2) for the *high* vibration condition and 'uncomfortable' for *low* vibration ($0.8 < a_T < 1.6$ m/s^2). Accelerometer sampling on the seatpan confirmed *x, y, z* and total (a_T) acceleration values of 0.15, 0.05, 1.08, and 1.10 m/s^2 respectively for the *low* vibration condition, and 0.72, 0.07, 2.11 and 2.15 m/s^2 for the *high* vibration condition. These frequency and acceleration levels are much higher than previous studies of ship motion (e.g., [25, 41]).

Trackball input was provided through a Logitech M570 device, which includes a scrollwheel that was used for controlling zoom-level in the pan-zoom tasks.

Participants
Eighteen volunteer participants were recruited for the study – 11 female, aged 19-40 (mean 27). All were familiar with touchscreen input methods, reporting a mean daily use of ~2 hours. Participation in the experiment lasted for 1 hour, and was compensated with a $10 payment.

Design
Data from each task type was separately analyzed for dependent measures *trial time* and *error rate.*

Target selection data was analyzed using a 3×3×3×2 repeated-measures analysis of variance (RM-ANOVA) for factors *vibration* (none, low, high), *device* (touch, stencil, trackball), *distance* (96, 256, 512 px) and *size* (32, 64 px). *Keypad* and *Pan-zoom* data was analyzed using a 3×3 RM-ANOVA for factors *vibration* and *device. Dial* and *Slider* data was analyzed using a 3×3×4 RM-ANOVA for factors *vibration, device,* and *resolution* (4, 8, 32, 128 items for Dial; 2, 8, 32, and 128 items for Slider).

RESULTS
The 18 participants produced data for a total of 7244 successful trials across the five task types. Only 5 trials were abandoned (0.07%), all in high vibration.

Target Selection Tasks

Selection time analysis
RM-ANOVA showed a significant effect of *vibration* level on error free selection time (Figure 4): $F_{1.33,22.5} = 45.1$, $p < .001$, $\eta^2_G = 0.16$; Greenhouse-Geiser correction applied for sphericity violation. Mean selection times more than doubled as vibration increased from *none* (mean 1163 ms, sd 677), through *low* (1648 ms, sd 973), to *high* vibration (2360 ms, sd 1798). *Distance* had a significant effect on selection time ($F_{2,34} = 4.3$, $p < .05$), but the magnitude of its effect was small ($\eta^2_G = 0.009$), with mean times increasing through 1610, 1684 and 1860 ms for 96, 256 and 512 px distances. *Distance* was not a factor in any significant interactions.

Target *size* had a stronger and significant effect on selection time: $F_{1,17} = 139.7$, $p < .001$, $\eta^2_G = 0.12$. Mean selection times for small (32 px) and large targets (64 px) were 2137 and 1299 ms, respectively.

There was no significant main effect of *device*: $F_{2,34} = 1.15$, $p = .33$. Mean selection times were similar with *stencil* and *touch*, at 1648 and 1686 ms respectively, and slightly higher for *trackball* (1820 ms). However, there was a significant *device* × *size* interaction ($F_{2,34} = 3.46$, $p < .05$). This effect was largely caused by the relatively similar mean selection

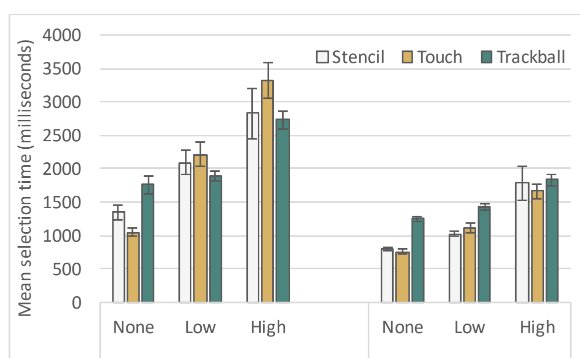


Figure 4. Error-free selection times for small (left) and large (right) targets by device and vibration. Error bars ±1 s.e.m.

times for the three devices with small targets (2089, 2193, and 2128 ms for *stencil*, *touch*, and *trackball* respectively), whereas with large targets, the trackball was slower (mean 1510ms) than *stencil* and *touch* (1207 and 1180 ms).

There was also a significant three way interaction of *device × size × vibration*: $F_{4,68} = 3.27$, $p < .05$. For small targets (Figure 4, left), when there was no vibration, mean selection times with *stencil* were slower than *touch*, but as vibration increased *stencil* selections became faster than *touch.* However, for larger targets (as shown in Figure 4, right), the benefits of the *stencil* over *touch* at high vibrations did not emerge; presumably because the targets were large enough to be easily acquired without a stabilizing edge. Also, the benefits of self-stabilization with the trackball become apparent at high levels of vibration, particularly with small targets. In summary, there is cross-over effect for small targets (that is largely absent with large targets) – unstabilized *touch* is fastest when vibration is absent, but it becomes slowest when vibration is present.

Error analysis

Two different interaction events could be interpreted as errors: 1. the user misses the target, instead hitting 'dead space' between active elements in the interface; 2. the user hits an incorrect target. Our error analysis focuses on incorrect target selections for two reasons. First, the stencil essentially eliminates the possibility for hitting interface dead-space because the holes are only present over active elements. Second, interactions on dead-space are likely to have lower consequences than incorrect selections. Incorrectly or accidentally hitting a target could have serious flight consequences, such as unknowingly changing a mode, whereas hitting dead-space has no effect.

Across all trials, an incorrect target was selected in 2.9% of trials. There were zero incorrect target selections with the trackball, even at *high* vibration.

There were a total of 45 errors with the *stencil* (rate 4.67%) and 40 with *touch.* Figure 5 summarizes error rate across conditions. One surprising observation from this analysis was that *stencil* errors were higher than *touch*.

Timing analysis showed the cause of this anomaly. It revealed that a large proportion of *stencil* errors were caused by a second selection of the target shortly after its correct selection. Figure 6 shows the cumulative proportion of *stencil* and *touch* errors occurring within 500 ms of a correct selection – 64% of stencil errors and 28% of touch errors occurred within 200 ms of correctly hitting a target. These errors occur during the 'lift off' action, and can be attributed to vibration causing accidental activation during 'lift off'. Dragging the finger up the stencil edge during lift-off appears to increase the incidence of these errors.

These stencil accuracy problems are easily remediated by discarding contacts that occur within 200 ms of a previous contact. Discarding these 'bounce' contacts within 200 ms of a previous touch reduces the overall mean error rate for the

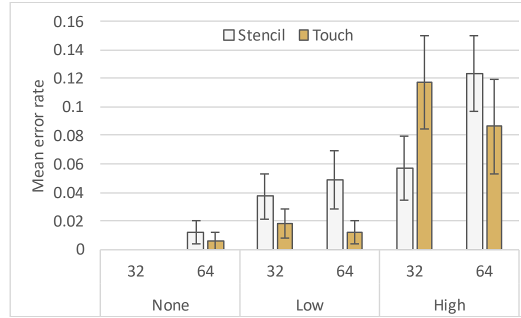


Figure 5. Error rates for *stencil* and *touch* for differently sized targets at each vibration level. Error bars show ±1 s.e.m.

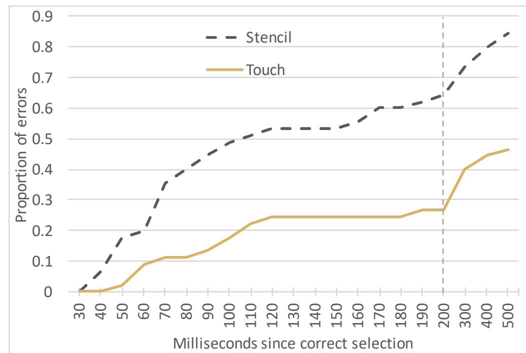


Figure 6. Cumulative proportion of errors with stencil and touch across time since correct selection.

stencil to 1.6% (and 2.8% for *touch*); and for small targets at high vibration, the *stencil* error rate was 2.5%, compared to 10.4% for *touch*.

To confirm that stencil errors can be eliminated by addressing 'bounce' errors on lift off, we conducted a small subsequent study of stencil and touch Target Selection with four participants at high vibration. We used the same stencil as the original study, but increased the target sizes to 54 (small) and 96 px (large) to make them easier to acquire with touch (subjective comments, reported later, indicated touch was too hard at high vibration). Each participant completed four repetitions of the same target pattern used in the original experiment, producing data for 576 trials (4 participants, 2 devices (stencil and touch), 4 repetitions, 18 targets). With the 200 ms delay between contacts, there were 11 incorrect target selections with *stencil* (3.8%), compared to 8 with *touch* (2.8%). If all touch contacts off the intended target were interpreted as errors (including all wrong target and dead-space selections), the total number of errors with *stencil* was 31 (mean 0.11 errors/selection) compared to more than six times as many with *touch* (198, 0.69 errors/selection).

Keypad Tasks

RM-ANOVA of keypad tasks showed significant effects of *device* ($F_{2,34} = 38.8$, $p < .001$) and *vibration* ($F_{2,34} = 20.0$, $p < .001$) on the mean time to enter and confirm a three-digit target number. There was no significant *device × vibration* interaction (p = .53). As shown in Figure 7, mean performance was very similar between *stencil* and *touch*

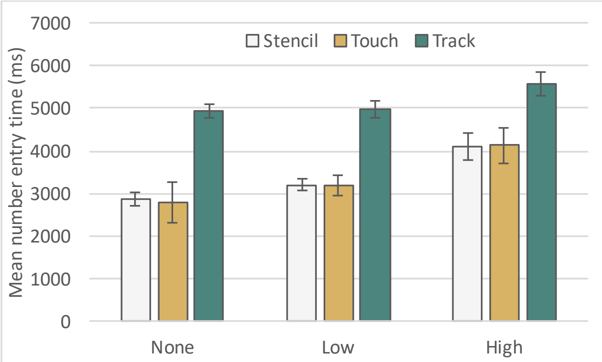


Figure 7. Mean time (± 1 s.e.) to enter a 3-digit number.

(3398 and 3371 ms, respectively), and *trackball* was substantially slower (5166 ms). The similarity of task time for *stencil* and *touch* suggests that users were able to confidently hit the large key targets without relying on the stencil edge, even when vibration was high.

Error rates were low. An incorrect number was submitted (by pressing the 'OK' button) 12 times from 542 tasks (2%): 6 times with *touch*, 4 times with *trackball*, and twice with *stencil*. The 'DEL' key was used to correct an incorrectly entered digit 48 times (from 1626 required digit entries) – 24 with *touch*, 4 with *trackball*, and 20 with *stencil.*

Slider Tasks

The performance of slider tasks was dominated by errors. Errors were high in all conditions, but particularly so for precise selections in high vibration settings. Figure 8 shows the mean number of errors per trial, which ranged from 0.019 for *trackball* selections with 2 candidate items and low vibration, through to a maximum of 1.97 for *touch* selections with 128 candidate items and high vibration. There were no significant effects on task time. Mean selection times were 1.9, 1.8 and 1.8 s with *stencil*, *touch* and *trackball*.

Even when there was no vibration, all of the devices produced high error rates (means of 0.37, 0.54 and 0.7 for *touch*, *trackball*, and *stencil*). The particularly high error rate for *stencil* (~1.5 per trial) with 128 items and no vibration may be due the finger contact forming unanticipated shapes (triggering unintended movement) during lift-off due to pressure against the stencil edge.

At high vibrations, the error rates were extremely high (overall mean of 0.9 errors per trial, ranging from 0.6 for *trackball* to 1.12 with *touch*). One factor contributing to these high error rates is the mechanical difficulty of maintaining contact pressure on a touch-surface (or trackball button) to continue the dragging state while undergoing vibration. The target selection and keypad tasks allowed users to make discrete 'stabs' at the surface or button, reducing the temporal window for vibrations to influence interaction. Dragging actions, in contrast, necessarily have a prolonged duration, increasing the opportunity for vibration to adversely affect execution.

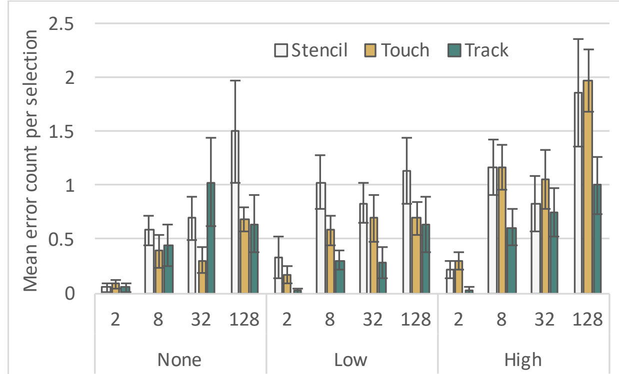


Figure 8. Mean errors (± 1 s.e.) per slider selection trial.

Setting values via sliders therefore seems to be an inadvisable interaction for a cockpit environment, where accuracy and error-free interaction is required.

Dial Tasks

Like slider tasks, dial tasks required maintaining a dragging state while manipulating the location of the arrow. Any unintended lift-off caused an error. In light of the slider results, it is therefore unsurprising that dial task performance was also dominated by errors, with an overall mean of 0.24 errors per trial. There were no significant effects on task time (means of 3.8, 3.5 and 3.4 s for *stencil*, *touch* and *trackball*).

Mean errors per trial were lower with the *trackball* (0.06) than *stencil* (0.34) or *touch* (0.32). At *high* vibrations with 128 candidate items, all devices had high error rates: 0.33 with *trackball,* 1.89 with stencil, and 1.69 with touch.

Pan-Zoom Tasks

Pan-zoom tasks are interesting because completing them on a touchscreen requires manipulating dragging actions (i.e., pinch to zoom and pan). However, unlike slider and dial tasks, lift-off actions are anticipated as part of successful task completion – users are expected to repeatedly clutch their pinching/depinching gestures to attain the required zoom level. Also, completing zoom actions with the trackball did not require maintaining a dragging state because zooming could be directly controlled by rotating the scrollwheel.

Figure 9 summarizes selection time results for the pan-zoom task. As indicated by the figure, there was a significant main effect of *device* ($F_{2,34} = 42.8$, $p < .001$), with *stencil* and *touch* having relatively similar performance (9.76 and 8.49 s respectively, s.d. 3.4 and 2.8 s), and *track* much slower at 14.5 s (s.d., 4.3). Vibration also had a significant effect on selection time ($F_{2,34} = 5.98$, $p < .01$), with mean times increasing through conditions of *none* (10.2 s, s.d. 3.8), *low* (10.7 s, s.d., 4.2) and *high* (11.9 s, 5.0). There was no *device × vibration* interaction (p = .3).

Subjective Responses

Analysis of subjective responses reinforce the objective data reported above. We used Friedman tests to analyze participants' ratings of physical workload and frustration (reported on 7-point Likert scales) for *target selection* tasks across *vibration* and *device.* As shown in Figure 10 (left), participants' ratings of physical workload increased significantly as *vibration* increased, from a mean of 2.35 at *no* vibration, through 3.83 at *low* vibration, to 4.56 at *high* vibration – $X^2 = 39.8$, $p < .001$. There was no significant effect of *device* on physical workload ($p = .09$). Frustration ratings (Figure 10, right) showed significant effects of *vibration* ($X^2 = 34.7$, $p < .001$) and *device* ($X^2 = 16.1$, $p < .001$) with *touch* having the highest overall frustration rating (4.0), followed by *stencil* (3.2) and *track* (2.9).

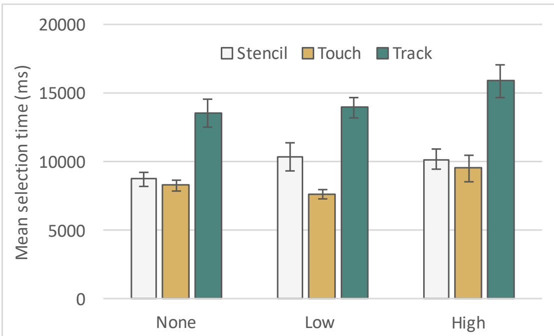


Figure 9. Mean selection times in pan-zoom trials for each input device at each vibration level. Error bars show ±1 s.e.m.

Participants' comments further emphasized that unstabilized *touch* interaction was difficult, whereas the *trackball* and *stencil* were easier to control in high vibration:

S1: "Stencil help to control where I want to point out on the screen … touchscreen is hardest to control especially for the small object."

S6: "Stencil was best because its borders offered a control or support for the tasks that were difficult under vibration (e.g., the tapping). The touchscreen was worst because even a little vibration threw me off."

S12: "The touchscreen was impossible in the 'tapping' run. My arm was out of my control … The stencil made it hard to mess up."

S13 "trackball [best for high vibration] because it was easiest, least jerky (i.e., most stable), most precise".

S18 "I found the stencil helps a lot when I want to press exact places. It's a combination of benefits of using touchscreen (handy) & improves the exactness."

Several participants commented that when vibration was absent, touch interaction was the easiest input method and that the trackball was slow and awkward.

S1: "trackball takes time to move the pointer to where I want"

S13 "trackball was slowest and least user-friendly."

One participant commented that the stencil was distracting:

S17 "the limitations in space [with the stencil] made it more difficult to concentrate on the task. It's like doing more things at once."

Participant Strategies

Participants were not instructed on how best to use each of the input devices, nor on how to stabilize their input. Although we had anticipated that participants would rest their fingers, palm, or hand on the stencil surface to stabilize their input, only one participant did so (S11). In hindsight, this is understandable – the Lexan sheet minimally influenced the appearance of widgets on the display, and consequently, participants were likely resistant to making contact with the sheet to avoid accidental activations. Explicit training that the Lexan sheet would prohibit contact registration may have influenced participants' strategies and improved performance with the stencil.

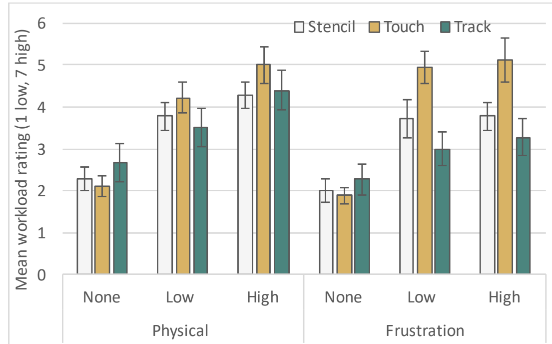


Figure 10. Mean responses (±1 s.e.) for physical workload and frustration by vibration (7 point Likert scales, 1 low, 7 high).

Most participants quickly learned that when vibration was present they needed to stabilize their hand and finger in order to complete tasks with *touch* and *stencil*. Typically they did so by placing their fingers or thumb on the screen edge (off the contact surface) and spanning their hand to make contact with the target on the display. Target selection tasks were located on the left side of the display (see the holes in Figure 1). Most users therefore placed the thumb of their right hand on the left or top display edge, and spanned to the targets with their index or middle finger. The same strategy was used for both *touch* and *stencil*.

Slider tasks were located near the bottom of the display (Figure 3). This placement allowed most participants to grasp the screen edge with their thumb and span their index finger to the slider widget. Keypad tasks were located near the right edge of the display, and some participants grasped the right edge of the display with the fingers of their right hand and spanned their thumb to the targets. However, keypad targets were sufficiently large for most participants to complete selections without stabilization.

DISCUSSION

Seven key findings emerge from the results:

1. There was a wide variance in the different tasks' overall tolerance to vibration. Tasks with low pointing precision requirements (e.g., Keypad) could be accomplished successfully with any device under all vibration conditions; others (particularly Slider) were difficult in all cases (even without vibration).
2. When pointing precision requirements were low, touch-based interactions (*stencil* or *touch*) were generally faster than trackball input. This includes the selection of large targets (e.g., the 104 px wide keys in the keypad task) and coarse pan-zoom actions, even at high vibration levels.
3. However, when selecting small targets in a vibrating environment, trackball input was faster than unstabilised touch-based input.
4. Trackball input was accurate, even at high vibrations.

5. Drag-based selections, in which lift-off/release confirmed selection, were inaccurate with touch and trackball. Errors were high when there was no vibration, and became extremely high when vibration was present.
6. The stencil was generally unsuccessful. The *device* × *size* × *vibration* interaction in target selection tasks (Figure 4) suggests that the *stencil* may improve touch interaction with small targets at high vibrations, but further work is needed to confirm this effect. Also, the stencil relies on a timeout after each selection (~200 ms), without which it is susceptible to finger lift-off errors during vibration. Froehlich et al. [11] used a similar timeout method to eliminate 'bounce' errors by users with tremor.
7. Multi-touch pan and zoom selections were much faster than equivalent interactions with a trackball (panning and scrollwheel zooming) at all vibration levels.

Touch vibration tolerance

Prior to conducting the study, we suspected that participants would have extreme difficulty in completing tasks in the *touch* condition at high vibration. However, results showed that participants were resilient, maintaining their ability to interact by stabilizing their hand on the screen edge.

Our experimental conditions facilitated this hand stabilization because the smallest targets (and therefore hardest to acquire) were located near the screen edges. Based on our observations of strong reliance on hand stabilization, we suspect that performance with *touch* would have been much slower and more error prone if the target widgets had been placed nearer the center of the display. Central placement would reduce the availability (or effectiveness) of the hand-spanning strategy we observed, with the fingers or thumb placed on the screen edge.

However, flight displays could easily be constructed to facilitate hand stabilization. Each display could be small, or on a large display, touch-inactive horizontal and vertical bezel ridges for finger/thumb placement could be overlaid on the display, with the distance between the ridges configured to permit easy hand-spanning to targets. There are also interesting opportunities for augmenting touchscreens with graspable or tactile widgets that could be positioned on or near the display (e.g., [14, 19, 28, 34, 35]).

Lessons from the stencil

Results from the *stencil* are interesting, suggesting both strengths that can be deployed, and weaknesses that should be avoided. In terms of strengths, several participants referred to the intended benefits of stabilization. Also, while raw wrong-target errors were increased by the stencil, timing analysis indicated that once lift-off errors were eliminated (by disabling touch for 200 ms following a selection), the stencil had similar wrong-target error rates as unstabilised touch, and dramatically fewer off-target errors.

However, participants seemed reluctant to place their fingers or hand onto the stencil surface, possibly because they mistakenly believed that their contacts would be registered there. Additionally, at least one participant referred to visual 'distraction' caused by the overlay. For these reasons, it seems that the extra bezel ridges described above, may be preferable to the transparent overlay examined in our study.

Study limitations and further work

To our knowledge, this is the first study of touch interaction in cockpits that covers a range of important task types and different vibration levels, as well as examining a potential remediation technique. As touchscreens are increasingly used in aircraft, there is need for further research on the problems of turbulent input and their remediation.

Our study used a limited range of vibration profiles, partially due to the maximum displacement (150 mm) attainable by our motion platform. Further studies could examine different vibrations, such as higher amplitudes and varied frequencies. Another obvious area for further study concerns the impact of target distribution on performance with the various devices (e.g., sparsely versus densely packed targets).

Participants in our study were predominantly university students. While this demographic is familiar with touchscreen interaction, none were pilots, and they were not explicitly trained on how to best exploit the evaluated technologies. There are therefore opportunities to study whether comparative user performance with stencils and trackballs changes when users are well trained. Similarly, testing with real pilots and real cockpit display systems will help validate findings for commercial use. Also, our trackball input used default control-display gain functions provided by the device manufacturer, which are likely to differ from the proprietary functions used by aircraft manufacturers.

Finally, we plan to test these results in other settings where touchscreens are being introduced into environments that can experience turbulence. For example, touchscreens are becoming common in cars and motorbikes – and here, the added concern of visual attention becomes an important issue. In further studies we will test whether the tactile feedback provided by the stencil reduces the amount of time users must attend to the display during vibration episodes.

CONCLUSION

Touchscreen interaction in the cockpits of commercial aircraft offers potential advantages to aircraft manufacturers, airlines, and pilots. However, turbulence is a challenge for their deployment. Our studies of touch interaction in simulated turbulence indicate that users can successfully stabilize discrete touch input (i.e., target selections) by touching the side of the screen with their fingers or thumbs. For small targets, a stencil overlay assisted users in hitting targets. Drag-based selections that confirmed selection on lift-off/release were highly error prone during simulated vibration, both with touch and trackball. However, multi-touch pan-and-zoom selections were much faster with touch than trackball.

REFERENCES

1. ARINC661 Cockpit Display System Interfaces to User Systems. ARINC Specification 661, AEEC, 2002.
2. ARINC661 Cockpit Display System Interfaces to User Systems ARINC Specification 661, supplement 6, AEEC, 2016.
3. Avrahami, D. The Effect of Edge Targets on Touch Performance. Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, ACM, Seoul, Republic of Korea, 2015, 1837-1846.
4. Barbé, J., Chatrenet, N., Mollard, R., Wolff, M. and Berard, P. Physical ergonomics approach for touch screen interaction in an aircraft cockpit. Proceedings of the 2012 Conference on Ergonomie et Interaction homme-machine, ACM, Biarritz, France, 2012, 9-16.
5. Bauersfeld, K.G. Effects of turbulence and activation method on touchscreen performance in aviation environments. Psychology, San Jose State University, 1992.
6. Buxton, W., Hill, R. and Rowley, P. Issues and techniques in touch-sensitive tablet input. SIGGRAPH Comput. Graph. 19, 3 (1985), 215-224.
7. Cockburn, A., Ahlström, D. and Gutwin, C. Understanding performance in touch selections: Tap, drag and radial pointing drag with finger, stylus and mouse. International Journal of Human-Computer Studies 70, 3 (2012), 218-233.
8. Dodd, S., Lancaster, J., Miranda, A., Grothe, S., DeMers, B. and Rogers, B. Touch Screens on the Flight Deck: The Impact of Touch Target Size, Spacing, Touch Technology and Turbulence on Pilot Performance. Proceedings of the Human Factors and Ergonomics Society Annual Meeting 58, 1 (2014), 6-10.
9. Fletcher, M. and Huffman, D. Panoramic cockpit displays for tactical military cockpits. Javidi, B., Son, J.-Y., Thomas, J.T. and Desjardins, D.D. eds. Proc. SPIE 7690, Three-Dimensional Imaging, Visualization, and Display 2010 and Display Technologies and Applications for Defense, Security, and Avionics IV, 769016, Orlando, Florida, 2010, 769016--769016-769017.
10. Forlines, C., Wigdor, D., Shen, C. and Balakrishnan, R. Direct-Touch vs. Mouse Input for Tabletop Displays. in Proceedings of the 25th international conference on Human factors in computing systems, ACM, (2007), 647-656.
11. Froehlich, J., Wobbrock, J.O. and Kane, S.K. Barrier pointing: using physical edges to assist target acquisition on mobile device touch screens. Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility, ACM, Tempe, Arizona, USA, 2007, 19-26.
12. Hamon, A., Palanque, P., Andrea, R., Barboni, E., Cronel, M. and Navarre, D. Multi-Touch interactions for control and display in interactive cockpits: issues and a proposal. Proceedings of the International Conference on Human-Computer Interaction in Aerospace, ACM, Santa Clara, California, 2014, 1-10.
13. Hardinger, C.E. Built to touch. Rockwell Collins Horizons. Vol. 17(1). 2011, 18-19.
14. Harrison, C. and Hudson, S.E. Providing dynamically changeable physical buttons on a visual display. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, Boston, MA, USA, 2009, 299-308.
15. Hill, S. and Tauson, R. Soldier performance issues in C2 "On the move". 10th International Command and Control Technology Symposium, McLean, VA, USA, 2005.
16. Hourlier, S., Guérard, S., Barou, J.L. and Servantie, X. Testing Touch Screens in Realistic Aeronautic Turbulent Conditions (Light to Severe). SAE International Journal of Aerospace 8, 2 (2015), 243-247.
17. Huseyin, A., Fischer, J.E. and Rodden, T. Designing touch-enabled electronic flight bags in SAR helicopter operations. HCI-Aero: International Conference on Human-Computer Interaction in Aerospace, ACM, Paris, France, 2016.
18. Huseyin, A., Fischer, J.E. and Rodden, T. Future Flight Decks: Impact of +Gz on Touchscreen Usability. HCI-Aero: International Conference on Human-Computer Interaction in Aerospace, ACM, Paris, France, 2016.
19. Jansen, Y., Dragicevic, P. and Fekete, J.-D. Tangible remote controllers for wall-size displays. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, Austin, Texas, USA, 2012, 2865-2874.
20. Kaminani, S. Human computer interaction issues with touch screen interfaces in the flight deck. in Digital Avionics Systems Conference (DASC), 2011 IEEE/AIAA 30th, (2011), 6B4-1-6B4-7.
21. Kolbe, D.M. Method for determining valid touch screen inputs. Patent: Application CA2792590A1. 2012. General Electric Company
22. Komer, J.L., Gepner, J.E., Hogan, R.K. and Mabie, T.D. Avionics control and display unit having cursor control mode of operation. Patent: US8626360B2. 2014. Garmin International Inc.
23. Levesque, V., Oram, L., MacLean, K., Cockburn, A., Marchuk, N.D., Johnson, D., Colgate, J.E. and Peshkin, M.A. Enhancing Physicality in Touch Interaction with Programmable Friction. in Proceedings of CHI'11: ACM Conference on Human Factors in Computing Systems, ACM, (2011), 2481-2490.

24. Lewis, R. The Effect of Handedness on use of Touch Screen versus Touch Pad. Proceedings of the 5th International Conference on Application and Theory of Automation in Command and Control Systems, ACM, Toulouse, France, 2015, 115-120.
25. Lin, C.J., Liu, C.N., Chao, C.J. and Chen, H.J. The performance of computer input devices in a vibration environment. Ergonomics 53, 4 (2010), 478-490.
26. Mack, R. and Lang, K. A Benchmark Comparison of Mouse and Touch Interface Techniques for an Intelligent Workstation Windowing Environment. in Proceedings of Human Factors Society 33rd Annual Meeting, (1989), 325-329.
27. McLeod, P., Poulton, C., Du Ross, H. and Lewis, W. The influence of ship motion on manual control skills. Ergonomics 23, 7 (1980), 623--634.
28. Miruchna, V., Walter, R., Lindlbauer, D., Lehmann, M., Klitzing, R.v., Johansson, E. and Müller, J. GelTouch: Localized Tactile Feedback Through Thin, Programmable Gel. Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, ACM, Daegu, Kyungpook, Republic of Korea, 2015, 3-10.
29. Narayanamoorthy, R. and Huzur Saran, V. Error analysis of task performance with laptop in vibration environment. Computers in Human Behavior 27, 6 (2011), 2263--2270.
30. Potter, R.L., Weldon, L.J. and Shneiderman, B. Improving the accuracy of touch screens: an experimental evaluation of three strategies. in Proceedings of the SIGCHI conference on Human factors in computing systems, ACM, (1988), 27-32.
31. Ren, X. and Moriya, S. Improving selection performance on pen-based systems: a study of pen-based interaction for selection tasks. ACM Trans. Comput.-Hum. Interact. 7, 3 (2000), 384-416.
32. Robinson, T., Grabski, G., Green, J., Jacobson, M., Byrne, C. and Harper, D. Physical Touch Aspects of the Touch Interface for Flight Deck Applications. SID Symposium Digest of Technical Papers, 2014, 1618–1621.
33. Stanton, N., Harvey, C., Plant, K. and Bolton, L. To twist, roll, stroke or poke? A study of input devices for menu navigation in the cockpit. Ergonomics 56, 4 (2013), 590-611.
34. Tuddenham, P., Kirk, D. and Izadi, S. Graspables revisited: multi-touch vs. tangible input for tabletop displays in acquisition and manipulation tasks. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, Atlanta, Georgia, USA, 2010, 2223-2232.
35. Ullmer, B., Dell, C., Gil, C., Cornelius Toole, J., Wiley, C., Dever, Z., Rogge, L., Bradford, R., Riviere, G., Sankaran, R., Liu, K., Freeman, C., Alvin Wallace, J., DeLatin, M., Washington, C., Reeser, A., Branton, C.W. and Parker, R. Casier: structures for composing tangibles and complementary interactors for use across diverse systems. Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction, ACM, Funchal, Portugal, 2011, 229-236.
36. Vogel, D. and Baudisch, P. Shift: A Technique for Operating Pen-Based Interfaces Using Touch. in Proceedings of CHI'07: ACM Conference on Human Factors in Computing Systems, ACM Press, (2007), 657-666.
37. Wang, L., Cao, Q., Chang, J. and Zhao, C. The Effect of Touch-key Size and Shape on the Usability of Flight Deck MCDU. The Eighth International Conference on Advances in Computer-Human Interactions, IARIA, Lisbon, Portugal, 2015, 234-238.
38. Wobbrock, J., Chau, D. and Myers, B. An Alternative to Push, Press, and Tap-tap-tap: Gesturing on an Isometric Joystick for Mobile Phone Text Entry. in Proceedings CHI 07: ACM conference on Human factors in computing systems, ACM, (2007), 667-676.
39. Wobbrock, J. and Myers, B. Trackball Text Entry for People with Motor Impairments. in Proceedings of CHI'06 ACM Conference on Human Factors in Computing Systems, ACM Press, (2006), 299-308.
40. Wobbrock, J., Myers, B. and Kembel, J. EdgeWrite: A stylus-based text entry method designed for high accuracy and stability of motion. in Proceedings of the 16th annual ACM symposium on User Interface Software and Technology, ACM Press, (2003), 61-70.
41. Yau, Y.-J., Chao, C.-J. and Hwang, S.-L. Effects of Input Device and Motion Type on a Cursor-Positioning Task. Perceptual and Motor Skills 106, 1 (2008), 76-90.
42. Zammit-Mangion, D., Becouarn, L., Aymeric, B., Fabbri, M. and Bader, J. A Single Interactive Display Concept for Commercial and Business Jet Cockpits. in 11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, American Institute of Aeronautics and Astronautics, 2011.
43. Zhai, S., Kristensson, P.O., Appert, C., Anderson, T.H. and Cao, X. Foundational Issues in Touch-Surface Stroke Gesture Design — An Integrative Review. Foundations and Trends in Human–Computer Interaction 5, 2 (2012), 97-205.