

Estimating Touch Force with Barometric Pressure Sensors

Philip Quinn

Google

Mountain View, CA

philip@quinn.gen.nz

ABSTRACT

Finger pressure offers a new dimension for touch interaction, where input is defined by its spatial position and orthogonal force. However, the limited availability and complexity of integrated force-sensing hardware in mobile devices is a barrier to exploring this design space. This paper presents a synthesis of two features in recent mobile devices – a barometric sensor (pressure altimeter) and ingress protection – to sense a user’s touch force. When a user applies force to a device’s display, it flexes inward and causes an increase in atmospheric pressure within the sealed chassis. This increase in pressure can be sensed by the device’s internal barometer. However, this change is uncontrolled and requires a calibration model to map atmospheric pressure to touch force. This paper derives such a model and demonstrates its viability on four commercially-available devices (including two with dedicated force sensors). The results show this method is sensitive to forces of less than 1 N, and is comparable to dedicated force sensors.

CCS CONCEPTS

• **Human-centered computing** → Ubiquitous and mobile computing systems and tools; *Touch screens*; • **Hardware** → Sensors and actuators.

KEYWORDS

Touchscreen interaction; touch pressure; sensor fusion.

ACM Reference Format:

Philip Quinn. 2019. Estimating Touch Force with Barometric Pressure Sensors. In *CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019)*, May 4–9, 2019, Glasgow, Scotland UK. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3290605.3300919>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

CHI 2019, May 4–9, 2019, Glasgow, Scotland UK

© 2019 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5970-2/19/05.

<https://doi.org/10.1145/3290605.3300919>

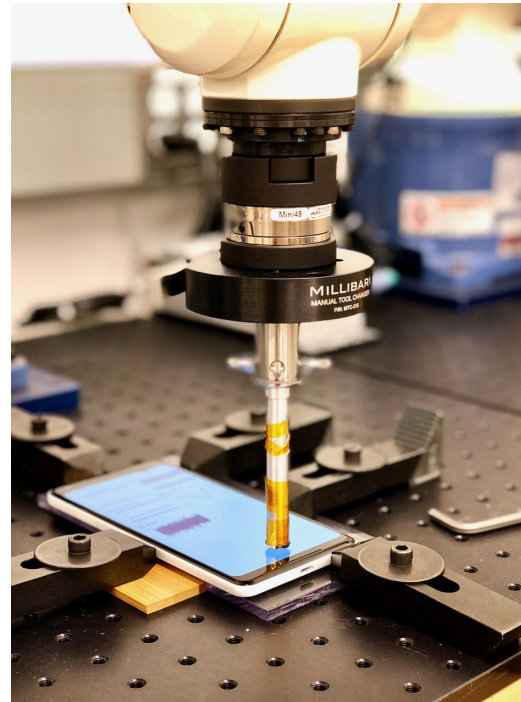


Figure 1: A Google Pixel 2 XL device being characterised with a DENSO VP-6242G six-axis robotic arm.

1 INTRODUCTION

Touch-based interactions consist of both a spatial location and a contact force. Even when force is not a parameter of the interaction, users will inevitably perform a force profile with their contact [24]. However, although force is an integral component of many stylus gestures [e.g. 17, 18, 32], it is not prevalent in touch gestures. A substantial factor in this disparity is the limited availability of force-sensing hardware for designers to prototype and experiment with compelling interactive experiences that utilise contact force.

Most commercially-available mobile devices feature barometers (pressure altimeters) to aid navigation and fitness tracking features, and it is becoming common for devices to also feature dust and water ingress protection. To guard against dust and water ingress a device must be sealed and almost airtight – isolating the internal atmosphere of the device chassis

from the external environment. Once a device is sealed, pushing on its external flexible surfaces (e.g. the display) is akin to squeezing a balloon, and produces an increase in the internal atmospheric pressure. This change in internal pressure can be measured using the barometer before it equalises with the external atmosphere.

In order to utilise this signal the relationship between the forces applied to different locations on the display and internal atmospheric pressure changes must be characterised. This also establishes the limitations of the technique in terms of the types of force it can accurately sense (relative vs. absolute, minimum resolution, etc.). In this paper we examine Google Pixel 2 devices and Apple iPhone devices to characterise their barometric responses to forces on their display, and develop a model to map barometric sensor output to an estimate of touch force. The latter devices also feature dedicated force sensors, which allow us to validate the accuracy of this sensing technique.

2 BACKGROUND

The most common and commercially-mature methods for sensing the forces applied to a surface are Force Sensing Resistors (FSRs) and strain gauges, which change their electrical resistance with the forces upon them [21, 22, 31]. A set of interleaved electrodes (FSRs) or discrete sensor elements (strain gauges) are arranged over a surface to measure its deformation from forces upon any part of it. When combined with a touch sensor that can resolve the spatial location of a contact, a factory calibration procedure and interpolation algorithm can produce accurate touch-force sensing.

Force sensing without additional sensor elements is also possible with certain capacitive touch-sensing methods [27]. For example, charge-sensing touch technologies allow one electrode of the sensor to ‘float’ above the other, fixed electrode [28]. When force is applied to the floating electrode, its distance to the fixed electrode is reduced, which causes the mutual capacitance between them to increase.

However, force sensing is not a standard feature of touch sensors, nor is it a pervasive feature on commercially-available mobile devices. Researchers have therefore explored augmented hardware, wearable hardware, or alternative sensing methods for detecting a user’s touch force. These include device cases with embedded FSRs/strain gauges [15], finger-mounted sensors [12, 29], and correlations with inertial sensors [8, 9, 11, 13, 16].

Interaction researchers have used these tools to explore the force component of touch input as an interaction parameter in a diverse design space [6, 10]. As with spatial position, force can either be treated as a continuous input for analogue parameter control (such as the speed of scrolling [2, 3] or zooming [8, 15, 19, 23]), or discretised into several levels for selecting between modes of operation (such as opening

context menus [9, 30], switching between keyboard layouts [5, 8], or as part of a gesture vocabulary [20]).

3 BAROMETRIC PRESSURE SENSING

Two trends in mobile device design have created a possible sensor-fusion approach for estimating touch force without a dedicated force sensor: (1) the inclusion of a barometer to sense atmospheric pressure for estimating altitude, and (2) enclosures with ingress protection against dust and liquid.

A barometer (pressure altimeter) measures the weight of the air above a sensing element (in N/m^2 – Pascals: Pa). Although atmospheric pressure can be related to absolute altitude [4], local variations make such sensors more suitable for measuring relative altitude (at $\approx 0.12 \text{ hPa/m}$). Barometers have been featured as standard sensors on mobile devices since the Samsung Galaxy Nexus (2011), and are commonly used for navigation and fitness tracking features.

Ingress protection guards against the intrusion of dust or liquid into a device chassis. The level of protection is typically measured against IEC standard 60529 – for example, a device rated with an IP67 code protects against the ingress of any dust, or water at pressures sustained at a depth of 1 m for 30 min. This is achieved using rubber/silicone gaskets and adhesive sealing around joins, gaps, or exposed components (e.g. around the display, physical buttons, connector ports, and speakers) – creating a boundary between the external atmosphere and that internal to the device chassis.

However, creating a completely impermeable boundary is not desirable. If the device were completely airtight, then the pressure on the device’s internal components would vary with altitude and temperature.¹ For example, transporting a device on an aircraft (at a lower atmospheric pressure than at sea level) would place a large amount of stress on the seals and internal components as the pressure differential tries to tear the device apart.

To resolve this, barometric vents are added to allow atmospheric pressure differences to equalise. The vents are created from an expanded polytetrafluoroethylene (PTFE) membrane, which is naturally hydrophobic, and with a structure to prevent penetration by dust particles. The membrane allows for a choked flow of air (measured in ml/min/cm^2) to slowly equalise the internal chassis pressure with that of the external atmosphere.

These two features (the barometer and ingress protection with venting) create an environment where sudden changes to the atmospheric pressure inside a device chassis can be readily sensed by its barometer – which slowly equalises over a short period of time through the barometric vent. In particular, placing a force on the device’s glass display will flex it inwards, reducing the internal volume, and creating a

¹It would also render an internal barometer functionally useless.

small increase in the atmospheric pressure within the device (akin to squeezing a balloon). Conversely, releasing this force will have the opposite effect on the internal pressure relative to the external atmosphere.

This creates a simple but effective mechanism for measuring touch force on a device near the moment of contact and tracking it as the internal pressure equalises. Although this is not a generalised force sensor, it can be a useful tool for researchers and designers to experiment with force-based interactions or to prototype specialised hardware when a dedicated force sensor is prohibitive.

Changes in air pressure have also been used as a sensing mechanism for other interactive techniques [e.g. 1, 14], and Takada et al. [25] also observed these effects for touch interaction on mobile devices – presenting a simple method for discriminating coarse, relative touch forces. We further cultivate the technique by deriving a physical model for a device's pressure equalisation that allows high-resolution continuous tracking of user-independent touch forces.

Modelling Pressure Equalisation

When a user applies a force to a device's display, the internal air pressure will follow a predictable pattern: it will rise commensurate with the force, and then decay as it equalises through the barometric vent.

This decay can be modelled from an application of the ideal gas law,

$$PV = nRT,$$

where P , V , n , and T are the pressure, volume, amount (moles), and temperature of the gas, respectively, and R is the ideal gas constant. The amount n can be written in terms of the gas' density ρ from its total mass m and molar mass M (with specific gas constant R_s):

$$\begin{aligned} n &= m/M, & \rho &= m/V, & R_s &= R/M, \\ P &= \rho R_s T. \end{aligned} \quad (1)$$

In this case the air is a compressible flow whose pressure P changes over time t due to a decrease in ρ from flow through the vent, modelled by the mass flow rate \dot{m} [7, 26]:

$$\frac{dP}{dt} = \frac{d\rho}{dt} R_s T, \quad (2)$$

$$\frac{d\rho}{dt} = \frac{\dot{m}}{V} = \frac{dm}{dt} V^{-1} = \frac{-\rho(t)Au}{V}, \quad (3)$$

where A is the area of the vent and u is the flow velocity.² For a choked isothermal flow (i.e. constricted by the size of the vent, and with no change in temperature),

$$u = \sqrt{P/\rho} = \sqrt{R_s T}.$$

²Conversely, when a user releases their force the change will be positive.

Therefore,

$$\rho = \frac{P}{R_s T}, \quad (\text{from 1})$$

$$\frac{d\rho}{dt} = \frac{-P(t)A\sqrt{R_s T}}{R_s TV}, \quad (\text{from 3})$$

$$\frac{dP}{dt} = \frac{-P(t)A\sqrt{R_s T}}{V}, \quad (\text{from 2})$$

$$P(t) = P_0 \exp \left[\frac{-A\sqrt{R_s T} \cdot t}{V} \right], \quad (4)$$

where P_0 is the initial gas pressure.³ As A , R_s , T , and V are effectively constants for a given device, their combined factors can be modelled as simply an exponential decay:

$$P(t) = P_0 \exp[-\lambda t]. \quad (5)$$

The parameter λ characterises a device's pressure equalisation, and deviations from Equation 5 can be used to measure changes in the user's applied force (described later).

4 DEVICE CHARACTERISATION

The barometer included in mobile devices is designed to measure external atmospheric changes, and not localised changes in internal pressure (which is an undesirable side-effect of the ingress protection). This means the placement of the sensor within a device, the cavities between components, and the location/size of barometric vents may not be optimal for sensing pressure changes from touch forces on the display. Similarly, the flex of a device's display is not even across its surface (e.g. pushing at the centre of a display will flex it more and create a larger increase in internal pressure than at its edges) and will vary with its material, shape, and attachment to the device chassis.

These issues require that each device is characterised and calibrated with respect to its barometer's sensing capabilities for touch forces.

We therefore characterised the responsiveness of device barometers to touch force on four commercially-available devices: the Google Pixel 2 and Pixel 2 XL (two of each), and the Apple iPhone 7 and iPhone X (one of each). The latter two also feature dedicated hardware for sensing touch force, facilitating comparison with barometer-based sensing.

Apparatus

A DENSO Robotics VP-6242G six-axis articulated robot arm performed controlled interactions on the devices (Figure 1). A metal stylus with a 3 mm thick, 80 mm diameter Shore A-40 hardness EPDM rubber tip and a grounded aluminium foil cover simulated a finger's touch contacts. The stylus was mounted on a load cell with a closed-loop PID controller for precise force control.

³In this case, the difference relative to the external pressure.

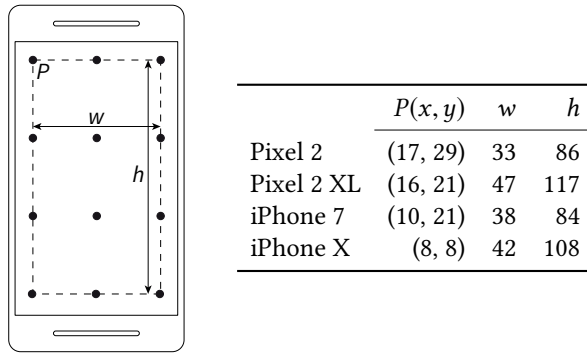


Figure 2: The contact locations on each device. Four rows and three columns were evenly spaced in the rectangle defined by an offset P (measured from the top/left edges of the device), and the lengths w and h (all in mm).

Procedure

The robotic arm was programmed to perform touch contacts at 12 locations on each device’s display (Figure 2). At each location the stylus pressed orthogonally to the screen at three force levels (1, 3, and 5 N) – each held for 5 s and then released for 5 s before moving to the next level/location.

Each device was tested three times, with all touch and barometric data recorded using its internal sensors.

Results

Figure 3 shows a representative sample of the pressure profile collected at each contact location. We used the pressure value 100 ms before a touch contact as a baseline to calculate the pressure delta during the contact. In general, after a force was applied to the display there was a sharp rise in barometric pressure commensurate with the force, followed by a decay as the internal and external pressures equalised.

To fit Equation 5 we took the peak value after each contact as P_0 and applied it as a normalisation factor for the subsequent data. This allows the decay parameter λ to be compared between force levels and between devices. We then found λ using log-level linear regression (i.e. $\ln[P(t)] = -\lambda t$; as $P_0 = 1$). Table 1 shows the median values of λ across all forces and locations for each device type tested. As expected from the derivation of $P(t)$, λ is consistent between all locations and force levels within each device type (indicated by the small confidence intervals; any differences in volume V from the location are too small to be of consequence).

There was substantial variation in the scale of the pressure profile across the matrix of contact locations (i.e. the maximum pressure delta for each force at each location). Figure 4 shows a representative map of the maximum barometric pressure delta recorded at each location for a 5 N force. As

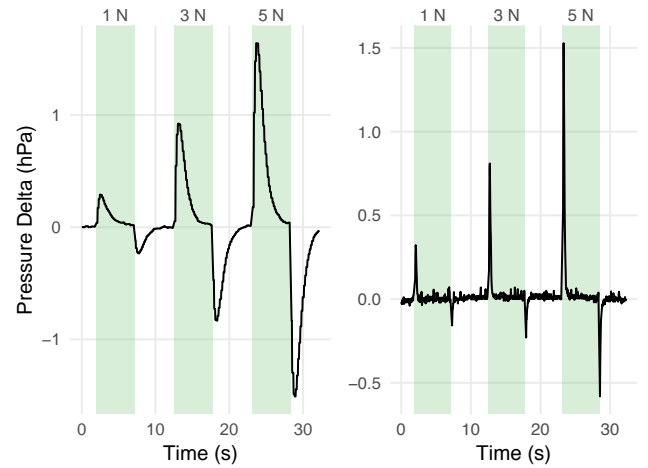


Figure 3: Barometric pressure profiles for a Google Pixel 2 (left) and an Apple iPhone X (right) for contacts of 1, 3, and 5 N (shaded) near the centre of the display.

Table 1: The median λ (and 95% CIs) across all locations and force levels, and the mean R^2 of the fit.

| | λ | | | R^2 |
|------------|-----------|-------|-------|-------|
| Pixel 2 | 0.76 | [0.72 | 0.80] | .96 |
| Pixel 2 XL | 1.06 | [0.98 | 1.11] | .96 |
| iPhone 7 | 6.72 | [6.45 | 6.92] | .91 |
| iPhone X | 7.01 | [6.74 | 7.38] | .91 |

expected, devices were most sensitive near the centre of their display and least sensitive near the edges.

For each device we fitted a linear model at each location to the relationship between the input force and the peak barometric pressure delta. As implied by Figure 4, the slope (β) of these models varied with location, but without adversely affecting the fit (mean $R^2 \geq .99$). Figure 5 shows the result of applying these models as force calibrators – illustrating the linearity of the response and the variance across test runs and locations. It is expected that the response will eventually saturate as the display meets rigid elements, but this was not apparent in the 1–5 N range on any device.

Sensor noise was low, with even the 1 N force detectable with a signal-to-noise ratio⁴ of between 5 (Pixel 2) and 10 (iPhone 7) at the weakest locations (the corners).

Force Sensor Comparison. The Apple iPhone 7 and iPhone X have dedicated touch-force sensors that can be tested and compared with barometer-based sensing using this method. They do not report force-sensor data as physical force, but

⁴Defined as the ratio of the 1 N peak signal to the standard deviation of the background (0 N) noise.

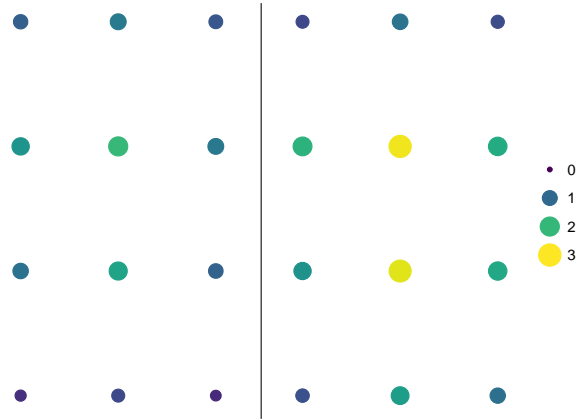


Figure 4: The maximum barometric pressure delta (hPa) for a 5 N force at each contact location (Figure 2) on a Google Pixel 2 XL (left) and an Apple iPhone X (right).

an arbitrary value $f \in [0, f_m]$, where $f = 1$ represents ‘the force of an average touch (predetermined by the system, not user-specific)’.⁵ Empirically, f_m appears to correspond to all forces above 5 N at a ‘firm’ sensitivity setting. The response of the force sensor was mostly linear, and the slope did not vary by location. The variance of the calibrated signal was comparable to barometer-based sensing (Figure 5).

5 TOUCH-FORCE SENSING ALGORITHM

The above characterisation provides a basis for an algorithm that converts touch input location and barometric pressure into a continuous measure of touch force. In particular, the linear relationship β between input force and peak barometric pressure is a model for converting changes in barometric pressure to changes in input force, and the decay λ is a model for tracking a baseline for barometric pressure over time.

For a given device, the linear slope β varies smoothly across the display (Figure 4), creating a surface that can be interpolated on to determine a β_p for a touch position p . The decay λ does not vary with location.⁶

If the input force is static, barometric pressure is expected to follow Equation 5 from any initial P_0 . Deviations between measured pressure $\hat{P}(t)$ and expected pressure $P(t)$ can be scaled by β_p to obtain the deviation in input force:

$$\Delta F = \beta_p [\hat{P}(t) - P(t)]. \quad (6)$$

When integrated, this provides a method for tracking smooth changes in input force. That is, the difference between the slopes of $\hat{P}(t)$ (empirical) and $P(t)$ (expected) is the rate that the input force is changing.

⁵<https://goo.gl/ww2Fjo>

⁶Depending on operating conditions, it may be necessary to factor temperature T out of Equation 4.

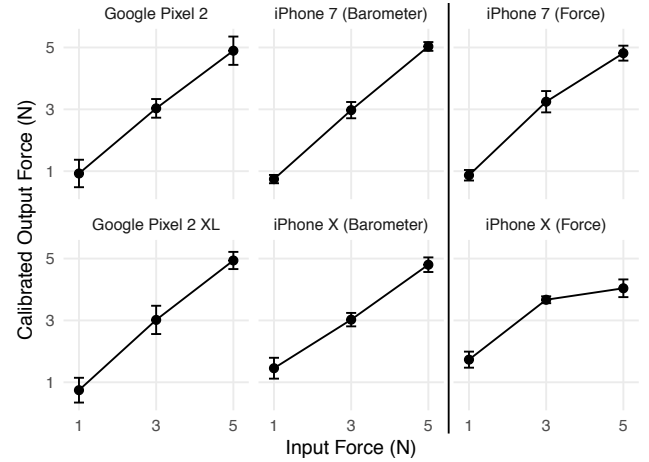


Figure 5: The mean calibrated output (\pm SD) at each tested input force of the barometric (left) and force (right) sensors across all locations.

However, as the input force is unlikely to change at a constant rate (i.e. users can modulate their force arbitrarily), a simple method to operationalise Equation 6 is to apply it discretely from local maxima. That is, using the pressure value shortly before a touch contact as a baseline, local peaks in pressure set P_0 and a force value F :

$$F = \beta_p P_0 + F^*,$$

where F^* is the calculated force value before the peak; initially 0. This force value F is adjusted by ΔF with each subsequent pressure sample above the noise floor (i.e. after a band-pass filter).

This algorithm allows for simple measurement of the instantaneous force coinciding with a contact (e.g. soft vs. hard taps), but also tracking the continuous modulation of force during a protracted contact (e.g. while scrolling or zooming).

6 DISCUSSION

These results demonstrate that measuring the atmospheric pressure within a sealed device chassis is an effective method for sensing forces applied to its exterior. Our characterisation of Google Pixel 2 and Apple iPhone devices found that although each device had different mechanical characteristics, they were consistent within themselves and followed an expected physical pressure model. All devices had the sensitivity to detect forces under 1 N, which can be continuously tracked using a simple algorithm.

The low noise and linear response are comparable with dedicated force sensors (Figure 5). The pressure decay (Equation 5) will accrue some tracking error when the internal/external pressure differential is close to zero (i.e. the signal approaches the noise floor), but relative changes in touch

force that provoke sharp changes in pressure are measurable with a high degree of accuracy. The consistency of responses between devices of the same type across test runs allows the algorithm's parameters (λ and β) to be reused for prototyping without per-device or per-user recalibration.

To fulfil their intended functions barometric sensors are rated to sense the relative change in atmospheric pressure corresponding to an altitude change of ≈ 1 m [4]. For the devices tested this sensitivity was well above that required to sense 1 N of force. The forces used in typical touch interactions are ≈ 0.5 N, but with substantial variability between tasks [24]. Force-based interactions are therefore likely to operate in the 1–5 N range we characterised.

Barometric sensors are resilient to inertial forces, and therefore a user's movement or handling of a device does not introduce interference. Although low-frequency environmental changes (e.g. temperature) will affect barometric pressure, they are accommodated by observing a baseline immediately before each contact and are unlikely to change during a touch contact.

However, the frequency response of this technique is bounded: on one side by the reported sampling rate of the barometer (we observed 25 Hz), and on the other by the flow rate of the barometric vent (i.e. λ). That is, if a user's input force changes either too quickly or too slowly, it may not be captured by the barometer due to under-sampling or rapid pressure equalisation, respectively. Similarly, if a large negative internal pressure is being equalised, touch contacts will need to overcome the pressure differential and λ for their force to be detected accurately.

Although this paper has focussed on forces upon the displays of mobile devices, this sensing technique is also applicable to other areas of a device and to other types of devices. For example, forces on the sides or back of a device will also yield changes in atmospheric pressure – however, without a corresponding sensor to confirm the presence of a user it may be difficult to attribute these changes to an intentional action. The broad availability and high sensitivity of barometric sensors also allows them to be embedded into novel input devices to measure the forces upon them.

ACKNOWLEDGEMENTS

The author thanks the Google Hardware Reliability Lab for their assistance with the device characterisation.

REFERENCES

- [1] Toshiyuki Ando, Yuki Kubo, Buntarou Shizuki, and Shin Takahashi. 2017. CanalSense: Face-Related Movement Recognition System Based on Sensing Air Pressure in Ear Canals. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, New York, NY, 679–689. <https://doi.org/10/cx9m>
- [2] Axel Antoine, Sylvain Malacria, and Géry Casiez. 2017. ForceEdge: Controlling Autoscroll on Both Desktop and Mobile Computers Using the Force. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, 3281–3292. <https://doi.org/10.1145/3025453.3025605>
- [3] Mathias Baglioni, Sylvain Malacria, Eric Lecolinet, and Yves Guiard. 2011. Flick-and-Brake: Finger Control over Inertial/Sustained Scroll Motion. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, New York, NY, 2281–2286. <https://doi.org/10.1145/1979742.1979853>
- [4] Mário N. Berberan-Santos, Evgeny N. Bodunov, and Lionello Pogliani. 1997. On the barometric formula. *American Journal of Physics* 65, 5 (1997), 404–412. <https://doi.org/10.1119/1.18555>
- [5] Stephen A. Brewster and Michael Hughes. 2009. Pressure-based Text Entry for Mobile Devices. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09)*. ACM, New York, NY, Article 9, 4 pages. <https://doi.org/10.1145/1613858.1613870>
- [6] William Buxton, Ralph Hill, and Peter Rowley. 1985. Issues and Techniques in Touch-sensitive Tablet Input. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques (SIGGRAPH '85)*. ACM, New York, NY, 215–224. <https://doi.org/10.1145/325334.325239>
- [7] Ron Darby and Raj P. Chhabra. 2017. *Chemical Engineering Fluid Mechanics* (3rd ed.). CRC Press, Boca Raton, FL.
- [8] Mayank Goel, Jacob Wobbrock, and Shwetak Patel. 2012. GripSense: Using Built-in Sensors to Detect Hand Posture and Pressure on Commodity Mobile Phones. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, 545–554. <https://doi.org/10.1145/2380116.2380184>
- [9] Seongkook Heo and Geehyuk Lee. 2011. ForceTap: Extending the Input Vocabulary of Mobile Touch Screens by Adding Tap Gestures. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. ACM, New York, NY, 113–122. <https://doi.org/10.1145/2037373.2037393>
- [10] Christopher F. Herot and Guy Weinzapfel. 1978. One-point Touch Input of Vector Information for Computer Displays. In *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '78)*. ACM, New York, NY, 210–216. <https://doi.org/10.1145/800248.807392>
- [11] Ken Hinckley and Hyunyoung Song. 2011. Sensor Synaesthesia: Touch in Motion, and Motion in Touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, 801–810. <https://doi.org/10.1145/1978942.1979059>
- [12] Min-Chieh Hsiu, Chiuan Wang, Da-Yuan Huang, Jhe-Wei Lin, Yu-Chih Lin, De-Nian Yang, Yi-ping Hung, and Mike Chen. 2016. Nail+: Sensing Fingernail Deformation to Detect Finger Force Touch Interactions on Rigid Surfaces. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, 1–6. <https://doi.org/10.1145/2935334.2935362>
- [13] Sungjae Hwang, Andrea Bianchi, and Kwang-yun Wohn. 2013. VibPress: Estimating Pressure Input Using Vibration Absorption on Mobile Devices. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '13)*. ACM, New York, NY, 31–34. <https://doi.org/10.1145/2493190.2493193>
- [14] Kyoungchul Kong and Masayoshi Tomizuka. 2009. A Gait Monitoring System Based on Air Pressure Sensors Embedded in a Shoe. *IEEE/ASME Transactions on Mechatronics* 14, 3 (2009), 358–370. <https://doi.org/10.1109/TMECH.2008.2008803>
- [15] Takashi Miyaki and Jun Rekimoto. 2009. GraspZoom: Zooming and Scrolling Control Model for Single-handed Mobile Interaction. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09)*. ACM, New

- York, NY, Article 11, 4 pages. <https://doi.org/10.1145/1613858.1613872>
- [16] Esben Warming Pedersen and Kasper Hornbæk. 2014. Expressive Touch: Studying Tapping Force on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, 421–430. <https://doi.org/10.1145/2556288.2557019>
- [17] Gonzalo Ramos, Matthew Boulos, and Ravin Balakrishnan. 2004. Pressure Widgets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04)*. ACM, New York, NY, 487–494. <https://doi.org/10.1145/985692.985754>
- [18] Gonzalo A. Ramos and Ravin Balakrishnan. 2007. Pressure Marks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, 1375–1384. <https://doi.org/10.1145/1240624.1240834>
- [19] Jun Rekimoto and Carsten Schwesig. 2006. PreSenseII: Bi-directional Touch and Pressure Sensing Interactions with Tactile Feedback. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems (CHI EA '06)*. ACM, New York, NY, 1253–1258. <https://doi.org/10.1145/1125451.1125685>
- [20] Christian Rendl, Patrick Greindl, Kathrin Probst, Martin Behrens, and Michael Haller. 2014. Presstures: Exploring Pressure-sensitive Multi-touch Gestures on Trackpads. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, 431–434. <https://doi.org/10.1145/2556288.2557146>
- [21] Ilya Rosenberg and Ken Perlin. 2009. The UnMousePad: An Interpolating Multi-touch Force-sensing Input Pad. *ACM Transactions on Graphics* 28, 3, Article 65 (2009), 9 pages. <https://doi.org/10.1145/1531326.1531371>
- [22] David A. Soss. 2007. Advances in Force-Based Touch Panels. *Information Display* 23, 12 (2007), 20–24.
- [23] Kenji Suzuki, Ryuuki Sakamoto, Daisuke Sakamoto, and Tetsuo Ono. 2018. Pressure-sensitive Zooming-out Interfaces for One-handed Mobile Interaction. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '18)*. ACM, New York, NY, Article 30, 8 pages. <https://doi.org/10.1145/3229434.3229446>
- [24] Faisal Taher, Jason Alexander, John Hardy, and Eduardo Velloso. 2014. An Empirical Characterization of Touch-Gesture Input-Force on Mobile Devices. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS '14)*. ACM, New York, NY, 195–204. <https://doi.org/10.1145/2669485.2669515>
- [25] Ryosuke Takada, Wei Lin, Toshiyuki Ando, Buntarou Shizuki, and Shin Takahashi. 2017. A Technique for Touch Force Sensing Using a Waterproof Device's Built-in Barometer. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. ACM, New York, NY, 2140–2146. <https://doi.org/10.1145/3027063.3053130>
- [26] James N. Tilton. 1997. Fluid and Particle Dynamics. In *Perry's Chemical Engineering Handbook* (7th ed.), Robert H. Perry, Don W. Green, and James O. Maloney (Eds.). McGraw-Hill, New York, NY, Chapter 6.
- [27] Geoff Walker. 2012. A review of technologies for sensing contact location on the surface of a display. *Journal of the Society for Information Display* 20, 8 (2012), 413–440. <https://doi.org/10.1002/jsid.100>
- [28] Geoff Walker and Mark Fihn. 2010. LCD In-Cell Touch. *Information Display* 26, 3 (2010), 8–14.
- [29] Yoichi Watanabe, Yasutoshi Makino, Katsunari Sato, and Takashi Maeno. 2012. Contact Force and Finger Angles Estimation for Touch Panel by Detecting Transmitted Light on Fingernail. In *Haptics: Perception, Devices, Mobility, and Communication*, Poika Isokoski and Jukka Springare (Eds.). Springer, Berlin/Heidelberg, 601–612. https://doi.org/10.1007/978-3-642-31401-8_53
- [30] Graham Wilson, Craig Stewart, and Stephen A. Brewster. 2010. Pressure-based Menu Selection for Mobile Devices. In *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '10)*. ACM, New York, NY, 181–190. <https://doi.org/10.1145/1851600.1851631>
- [31] S. I. Yaniger. 1991. Force Sensing Resistors: A Review Of The Technology. In *Electro International*. 666–668. <https://doi.org/10.1109/ELECTR.1991.718294>
- [32] Jibin Yin, Xiangshi Ren, and Shumin Zhai. 2010. Pen pressure control in trajectory-based interaction. *Behaviour & Information Technology* 29, 2 (2010), 137–148. <https://doi.org/10.1080/01449290902904733>