Optimizing Pressure Matrices: Interdigitation and Interpolation Methods for Continuous Position Input
Paul Strohmeier, Victor Håkansson, Cedric Honnet, Daniel Ashbrook, Kasper Hornbæk

To cite this version:
Paul Strohmeier, Victor Håkansson, Cedric Honnet, Daniel Ashbrook, Kasper Hornbæk. Optimizing Pressure Matrices: Interdigitation and Interpolation Methods for Continuous Position Input. ACM International Conference on Tangible, Embedded and Embodied Interaction (TEI), Mar 2019, Tempe, AZ, United States. 10.1145/3294109.3295638 . hal-01946451

HAL Id: hal-01946451
https://hal.archives-ouvertes.fr/hal-01946451
Submitted on 13 Jan 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Optimizing Pressure Matrices: Interdigitation and Interpolation Methods for Continuous Position Input

Paul Strohmeier\textsuperscript{1}, Victor Håkansson\textsuperscript{1}, Cedric Honnet\textsuperscript{2}, Daniel Ashbrook\textsuperscript{1}, Kasper Hornbæk\textsuperscript{1}
\textsuperscript{1}Human Centred Computing - University of Copenhagen, Denmark
\textsuperscript{2}Sorbonne University, CNRS, ISIR, Paris, France
\{p.strohmeier, dan, kash\} @ di.ku.dk, victor@skavi.dk, cedric@honnet.eu

ABSTRACT
This paper provides resources and design recommendations for optimizing position input for pressure sensor matrices, a sensor design often used in eTextiles. Currently applications using pressure matrices for precise continuous position control are rare. One reason designers opt against using these sensors for continuous position control is that when the finger transitions from one sensing electrode to the next, jerky motion, jumps or other non-linear artifacts appear. We demonstrate that interdigitation can improve transition behavior and discuss interpolation algorithms to best leverage such designs. We provide software for reproducing our sensors and experiment, as well as a dataset consisting of 1122 swipe gestures performed on 17 sensors.

Author Keywords
Sensor design; piezoresistive; pressure matrix; pressure input; interdigitation; interpolation; eTextile.

INTRODUCTION
Continuous multi-touch pressure and position input is becoming increasingly mainstream, notably finding application in synthesizer controllers [6,36,37]. Pressure based position input is also used in wearable systems, including on-body gesture input [29], user authentication [32], object recognition [35], control of IoT objects [22] or detection body deformations [23].

These systems are often implemented using a resistive pressure sensor matrix, which consists of a pressure sensitive layer surrounded by conductive strips in a row and column arrangement [5,6,23,29,35]. Often the sensor is connected to a gesture recognition system [23,29], allowing the user to perform discrete actions. Truly continuous position input, however, also requires locating pressure events that do not align with the sensor’s electrodes. This is typically achieved through interpolation [6,27], however, even with interpolation the transition behavior from one electrode-strip to the next introduces jumps or discontinuities [6,7]: Depending on the method chosen to extract the position of the finger, the signal might lag behind or jump ahead as a finger moves from one strip to the next (see video figure). There is no consensus if there is a best method to find touch-positions from such sensors, or what this method might be.

Our goal is to maximize the sensing resolution of pressure sensor matrices by methods other than increasing the electrode count. This can be done by leveraging the conductive properties of the resistive material [27], by interpolating the collected data [6], or by using diagonal or triangular electrode designs which create gradual transitions from one electrode to the next [34]. The latter approach, which we refer to as interdigitation, is quite common in DIY context (Figure 1, b,c,d,e), and can be found in product briefs and white-papers [15,30]. It is, however, not clear if and how such designs effect resistive pressure matrices.

We investigate effects of interpolation algorithm choice and electrode interdigitation on accuracy, precision, and consistency of touch-position sensing. In doing so we make three contributions: (1) We provide a data-set of 1122 swipe gestures performed on 17 sensors. This dataset is designed to

Figure 1 - a) Electrodes of two sensors used in this paper
b) capacitive sliders by Rachel Freire
c, d) electrode designs and circular slider by Kobakant
e) electrode testing samples at Datapaulette
enable quick testing of algorithms for extracting positions for various sensor designs, pressure levels and touch sizes. (2) We analyze and compare eight peak-detection algorithms for extracting touch position. (3) We investigate the effects of interdigitated patterns on sensor performance. In additionally, we also provide all code used in this project, including the pattern-generator used for creating the sensors and a prototype simulation tool for testing electrode patterns.

This project was started to optimize eTextile sensors. While the results generalize to any resistive pressure sensor matrix, we discuss them in the context of eTextile sensing.

**ADDITIONAL RESOURCES**

All code, data, and supplementary material can be found at: https://datapaulette.github.io/interdigitation/

**BACKGROUND AND RELATED WORK**

**Pressure Sensor Matrices**

Traditional resistive touchscreens can estimate the position of the touch, but not the amount of pressure exerted. There are variations which can do both, but these are uncommon, due to the complexity of their driving circuits [8]. A simpler alternative is to use a high number of discrete pressure sensors [16], however, this leads complex physical designs as $n$ pressure positions require at least $n^2$ physical connections. Pressure sensor matrices provide a middle ground as they are relatively simple to understand and implement, and also scale well: $n$ pressure positions only require $2\sqrt{n}$ physical connectors [8].

Pressure sensor matrices reduce physical complexity by multiplexing measurements; pressure sensitive strips are laid out in a row and column arrangement. Figure 2 (left) shows a schematic where the rows might connect to digital outputs and the column to analog inputs. The top left resistance is measured by grounding all rows, except for row 0 which is pulled high, and reading from column 0. The voltage values would then be sampled sequentially at all positions, resulting in a 2D array of rasterized pressure information (See Figure 2, center and right for a visualization of touch events).

**eTextile Touchpads**

Both the somewhat delicate mechanical layering required for designs inspired by traditional resistive touchpads [10] as well as the relatively fine traces of high resolution devices such as Rosenberg’s UnMousePad [27] are currently difficult to implement in fabric. As with conventional sensors, the workarounds include sensor arrays [17,21] that require a relatively large amount of physical connections. Various unconventional solutions, such as the XY textile by Donneaud [14] and a solution that shares many similarities with a 4-wire touchpad by Freed et al. [28] have been suggested, but are rarely seen as the relative simplicity of the pressure sensor matrix eventually made it the default design.

Due to their softness, pliability, and the ease with which they can be integrated in everyday objects such as clothing or furniture, textile pressure matrices are also commonly used for pose detection while sleeping [22] or sitting [36,40], as well as activity tracking [23,35,43]. Fabric pressure sensor matrices are also used for object recognition [27,42]. Within the HCI community, pressure sensor matrices have been used for gestural input [24,33] body pose as input [38] as well as user identification [39]. These applications do not require the detection of smooth transitions between strips. Instead, the maximum position is used with geometric models [33] or the analog data is converted to a binary image before applying computer vision methods [24,27,40]. Machine-learning methods are also used for extracting relevant information from the raw data [38,39]. The focus of this research lies on applications and the post processing of the sensor data. Only few papers report sensing accuracy [7] which makes it difficult to establish best practices for the physical design and construction of the sensor.

As pressure sensitive touch-surfaces are an intriguing musical input modality, the modern experimental music and NIME communities have been instrumental in pushing eTextile input forward [33]. The availability of piezo resistive materials supported prototyping of a range of new input methods [9] which eventually resulted in various pressure matrix designs both in the NIME [28] and DIY communities. Popular products such as the Roli Seaboard [36] or the Joué board [37] demonstrate the demand for high-resolution continuous input, which makes empirically establishing best practices increasingly relevant.

**Interpolation in Software and Hardware**

As pressure is sampled at discrete intervals, the center of an object or finger is typically not aligned with the sensing elements. Finding the center of a pressure point therefore requires additional steps (see Figure 2, center and right). Most work investigating this focuses on capacitive input, as it is currently the most common method for multi-touch sensing. Examples include comparisons of electrode patterns and interpolation methods [1].

Finding a signal maxima between sensing elements is common also outside of touch-screen design and is generally referred to as sub-pixel peak estimation [2,20]. An overview is presented by Naidu and Fisher [7,19] who conducted an empirical analysis of eight algorithms. They noted that all algorithms tested by them displayed periodic error patterns, similar to those observed in the transition behavior of textile pressure matrices [6]. Naidu and Fisher claim these errors to be “symptomatic of the sensor structure” [7].

There appear to be three factors influencing transition behavior: (1) the spread of the signal of interest [6,7], (2) the

![Figure 2](https://datapaulette.github.io/interdigitation/)

**Figure 2** - (left) 4 by 4 pressure matrix circuit. (Center) 16*16 pressure sensor matrix, and (right) raw data output [6].
peak detection algorithm [7] and (3) the physical design of the sensor [7]. However, there is comparatively little work exploring alternative sensor structures in pressure-sensor matrices. Rosenberg et al. present a matrix that utilizes the horizontal conductivity of the resistive material for better interpolation [25] and demonstrate how adding additional ‘dummy electrodes’ linearizes the horizontal conductivity [27]. The method used by Rosenberg et al. assumes a touch from the tip of a stylus. It is unclear if their approach generalizes to the larger touch area of a fingertip.

An alternative approach uses triangular interdigititation. If one moves a finger from one strip to the next at a fixed speed, the transition occurs faster if the motion is perpendicular to the strips than if the strips are arranged diagonally. Yoo and Pines demonstrate this approach in a ‘center of pressure’ sensor [34], however they chose a design which only supports input at discrete positions. Triangular interdigitated designs can also be found in white-papers and application notes for capacitive slider design [15,30] and are also common in the DIY community: They are being explored in Paris based textile hackerspace Datapaulette, they can be found in the web-archive by Satomi and Perner-Wilson [38], as Instructables by Freire [37] or in the eTextile swatchbook [11] (see also Figure 1). While triangular interdigitated designs are quite common, they are often justified based on intuition and – to the best of our knowledge – their effectiveness has never been evaluated.

SENSORS
Design
We created a processing sketch that generates interdigitated patterns and exports them as .pdf, ready for laser cutting (Figure 3). For the patterns used in the dataset we used triangular digits. We varied the digits using two parameters:

Digit Width: Low digit width allows many digits to fit (Figure 3, top). High digit width means that only few digits fit on the strip (Figure 3, bot). Digit Width is the dimension parallel to the sensing strip.

Digit Length: Low digit length means that there is little overlap between electrodes (Figure 3, left). High digit length means that there is significant overlap (Figure 3, right). Digit Length is the dimension perpendicular to the sensing strip.

Implementation
We built our sensors according to Donneaud et al. [6], however, we created the electrodes following instructions by Strohmeier et al. [31]: We used conductive ripstop [38] for creating the sensing electrodes. We prepared it by ironing the Ripstop to dual-sided fabric bonding. Then, using dual-sided scrapbook adhesive [39] we glue the ripstop to an MDF board (bonding glue side up). We insert the board with the ripstop into the laser-cutter and cut the pattern. Once we remove the MDF and ripstop from the laser cutter, we gently peel off all excess material from the board, so that only the electrodes remain glued to the board. We then place a second piece of textile – this time non-conductive – on top of the electrodes and iron it in place. Once the bonding has securely connected the electrodes to the non-conductive fabric, we pull both fabric layers off the MDF board.

This method allowed us to create more precise patterns, as we did not need to manually arrange the electrodes on fabric backing. We used piezo-resistive, non-woven fabric by Eeonyx (20kΩ per square) for our resistive layer (Figure 4).

Dimensions
We created 17 sensors. One sensor with no interdigititation, and 16 sensors with a 4*4 factorial design of width and length (digit length: 55%, 70%, 85%, 100% of strip spacing) and digit width: 95%, 75%, 55%, 35% of strip spacing). We chose our broadest digit (55% length, 95% width) to approximately match DIY designs and White Paper recommendations [15,30]. Our sharpest digit was limited by our manufacturing ability. We only created a single side of each matrix. As our primary interest is to better understand transition behavior between sensing electrodes, the full matrix is not required.

Each sensor was ~160mm wide and consisted of 7 electrodes with a 25mm spacing between strip-centers (the first and last strip were truncated). We use both larger touch-sizes and larger strip widths than we typically use. The larger strip width allows us to evaluate designs which we currently could not manufacture at fingertip scale, the larger touch sizes means we maintain the expected ratio between ‘finger’ and sensing elements. We chose the sizes so that the ‘small’ touch-size approximates the size of a human fingertip.
DATA COLLECTION
We used a CNC machine [40] to ensure consistent touch-behavior in terms of pressure and location. We created modified Dremel bits that acted as ‘fingers’ that would press on the sensor. The ‘fingers’ were made of Sugru with a rubber disk (cut from a mouse-pad [39]) beneath them to approximate the softness of the human fingertip (Figure 5).

Each strip was sampled by an analog input of an Arduino Uno. We wrote a JavaScript controller that coordinates data logging and communicates between the CNC and Arduino. The controller is open source and can be downloaded from the projects GitHub page. We used this controller to perform an approximation of a swipe gesture by measuring pressure at 70 consecutive locations on the sensor with 1mm spacing. For each sensor we repeated this swipe gesture 11 times, with 2.5mm offset for a total of 770 points.

This was repeated for three touch sizes (50%, 100%, 150% of strip spacing) crossed with two pressure settings (strong/gentle), where ‘strong’ was set to take full advantage of the dynamic range of the sensor and ‘gentle’ was the lowest pressure level we could consistently detect. In total we collect 4620 touch events per sensor for a total of 78,540 touch events over all 17 sensors.

Assumptions and Limitations
We report independent variables and results in percent of strip spacing (25mm center to center) to support generalization to sensors and patterns of different sizes, and comparisons to previous evaluations [6]. Another relevant unit, however, is the ratio of touch size to strip width. We assume that (within limits) there is an equivalency in increasing the touch size by 100% to reducing strip size by 50%. The touch-size conditions can be either thought of as changing the size of touch area, or as changing size of strip-width relative to a constant touch size.

Our application scenario is continuous control of audio or video content, so our primary goal is that the user’s motion over the eTextile results in a corresponding change in output signal, without jumps or discontinuities.

The Data
We provide the raw log files as we recorded them, including various metadata. We also provide a ‘normalized’ version. Here we removed the noise-floor, rescaled the strips so their dynamic range is from 0 to 1 and applied a low-pass filter.

All the modifications done to the ‘normalized’ data can be done in real time by an embedded system. For the rest of the paper we will be referring to the ‘normalized’ dataset.

ALGORITHM SELECTION
We chose eight algorithms for detecting touch-positions based on previous work on subpixel peak estimation [1,7] and precedent in eTextile matrix designs [6]. As a point of reference, we also add a ‘naïve’ estimator (NAIVE), which simply places the touch-position at the center of the strip with the highest pressure reading, as was used for gesture extraction by Schneegass et al. [29].

We test five algorithms that fit a function through the strip with the highest pressure reading and its neighbors. These assume that there is a linear (LINE) [1,7], gaussian (GAUSS) [1,7,20], cubic (CUBIC) [6,24], or parabolic (PARA) [1,7] spread of pressure from the maximum pressure point. Additionally we test a filtered version of the parabolic estimator, suggested by Blais and Rioux (B_R) [3,7].

We test two geometric solutions: The mTouch algorithm by Microchip [18] which fits a centroid around the maximum pressure values and its neighbors (mTOUCH) and the center of mass (COM) for all strips as a touch-estimator. For our sensor these are equivalent to COM3 and COM7 suggested by Fisher and Naidu [7].

These methods can further be combined with blob tracking: The function fitting methods can be used to calculate the boundaries of the blob [13], for example Donneaud et al. use bicubic interpolation to find the boundaries of blobs with greater accuracy [6]. The geometric methods can be used to improve the estimate of the blob center, for example Burstyn et al. detect blobs using the naïve approach, but estimate the touch-position within the blob using center of mass [4]. As these combinations also can be done at arbitrary threshold levels, testing them is beyond the scope of this exploration, however we believe our results can guide designers towards informed choices on reasonable combinations. We will refer to these algorithms as method from now on in the paper.

DATA ANALYSIS
Using the collected pressure values, we (1) compute the touch position for each measured position and each method. The resulting positions are in units of strip-width. We (2) scale each sensor*method combination to match the expected range in mm. We then (3) calibrate the values by subtracting the y intercept of the regression line and multiplying each value by one minus the slope of the regression line (Figure 6). The resulting positions are in the

Figure 5 - CNC rig used for collecting data.

Figure 6 – Data Processing steps
correct range, with a regression line of \( y = x \) (blue). We then subtract the known position from the estimated position to find the error. This resulted in 36,960 error measures per sensor (770 positions by 8 methods, 3 touch sizes, 2 pressure levels) for a total of 628,320 error measures.

We investigate sensor accuracy, precision and consistency over pressure levels, using measures derived from this error measure. For sensor accuracy we need a measure that describes how far the estimated positions are from the known position, on average. We therefore analyze the absolute error (36,960 data points per sensor). For sensor precision we wish to know how consistent the output is between swipes (3360 data points per sensor). We therefore analyze the standard deviation at each x-position of all swipe motions. Finally, for sensor consistency we wish to know if the output behavior is consistent across pressure levels. We therefore analyze the correlation between the mean swipe positions at strong and at gentle pressure levels (24 points per sensor).

While the multiple swipe measures might suggest using a repeated measures ANOVA, the swipes are all performed by the same device, and the variability between them is negligible. This would reduce our degrees of freedom without reducing the error term. We therefore opt for univariate ANOVAs. Our sample size for sensor accuracy is very high, allowing us to detect significant effects, even if the effect size is very small. Precision and consistency use successively higher-level concepts, leading to smaller samples sizes. For the consistency measure the sample size is so too small for a full factorial analysis.

For accuracy we analyzed absolute error by digit width, digit height, touch size, pressure and method. All main effects and all interaction effects were statistically significant at the p < .0001 level.

Precision was analyzed using standard deviation by digit width, digit height, touch size, pressure and method. All main effects and all interactions, except for touch size * method and digit width * method, were significant at p<.0001.

For consistency we performed two separate ANOVAs once correlation by digit width and digit height and once on correlation by touch size and method. There were no significant effects for digit width or digit height. For touch size and method, all main effects and all interaction effects were statistically significant at the p < .0001 level.

Due to our large sample sizes for accuracy and precision, it is not surprising that we could demonstrate significant differences even for very small effects. We therefore also report effect sizes, which typically were small, except for the effect of method, which was medium. The effect size reported is partial eta squared – any variance explained by other variables is removed. This allows for easy comparisons between parameters when exploring other electrode design manipulations. It should be noted that eta measures typically overestimate effect sizes, but that we believe this to be negligible in our case due to the large sample sizes.

All significant results maintain the p < .0001 level after correcting for family wise error – we will from now on not report these. The F-statistics and effect sizes, as well as Bonferroni corrected post-hoc analysis, will be presented where relevant.

**RESULTS**

### Pressure

<table>
<thead>
<tr>
<th>Pressure</th>
<th>Error</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
<th>Deviation</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gentle</td>
<td>10.86%</td>
<td>10.84%</td>
<td>10.89%</td>
<td>5.48%</td>
<td>5.40%</td>
<td>5.56%</td>
</tr>
<tr>
<td>Strong</td>
<td>11.63%</td>
<td>11.60%</td>
<td>11.66%</td>
<td>4.01%</td>
<td>3.94%</td>
<td>4.04%</td>
</tr>
</tbody>
</table>

**Table 1 – Means and 95% confidence intervals for pressure**

We found that on average that low pressure performed better than high pressure over all sensors for accuracy (F(1, 627504 = 627504, \( \eta_p^2 = .001 \)). This effect, however, is dwarfed by the comparatively larger effect of pressure on precision (F(1, 56304= 535.056, \( \eta_p^2 = .009 \)), where the strong pressure outperforms the gentle pressure. Increased accuracy of gentle touches comes at the cost of reduced precision.

### Touch Size

<table>
<thead>
<tr>
<th>Touch Size</th>
<th>Error</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
<th>Deviation</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>50%</td>
<td>11.96%</td>
<td>11.93%</td>
<td>12.00%</td>
<td>6.17%</td>
<td>6.17%</td>
<td>6.17%</td>
</tr>
<tr>
<td>100%</td>
<td>11.50%</td>
<td>11.46%</td>
<td>11.53%</td>
<td>4.68%</td>
<td>4.68%</td>
<td>4.68%</td>
</tr>
<tr>
<td>150%</td>
<td>10.28%</td>
<td>10.25%</td>
<td>10.32%</td>
<td>3.39%</td>
<td>3.39%</td>
<td>3.39%</td>
</tr>
</tbody>
</table>

**Table 2 – Means and 95% confidence intervals for touch size**

We vary touch size, as it allows us to explore various touch size to strip width ratios. One can also assume the 50%, 100% and 150% touch sizes to represent strip widths of 200%, 100% and 66.6% finger size.

Increasing the touch size had a positive effect on accuracy (F(2, 627504 = 1414.744, \( \eta_p^2 = .004 \)) and an even stronger effect on precision (F(2, 56304= 583.094, \( \eta_p^2 = .020 \)). Bonferroni corrected post-hoc analysis showed that each successively larger size performed significantly better than the previous size for both accuracy and precision (Table 1).

The interactions between strip width and touch size lead to various periodic behaviors, and some sensors display the opposite trend as we find on average. Figure 7 shows the average change between consecutive pressure readings for all x positions of a swipe gesture on the non-interdigitated sensor. For the small touch position (top) there is always a sensor where values are changing, providing information that the touch-point has moved between readings. For the medium touch size (middle), the areas where change occurs begin to cluster and for the large touch size (bottom) they begin to completely overlap. About half of the swipe distance there is no change in signal which might indicate a change in position (~10-25mm, 35-50mm, ~60mm), This causes most peak detection methods to first underestimate and then overestimate the change in position between measures.

The medium touch size performed best in terms of consistency over pressure levels (F(2, 384) = 15.26, \( \eta_p^2 = .074 \)).
Digit Length

<table>
<thead>
<tr>
<th>Digit Length</th>
<th>Error</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
<th>Deviation</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
<td>11.26%</td>
<td>11.17%</td>
<td>11.34%</td>
<td>2.43%</td>
<td>2.21%</td>
<td>2.65%</td>
</tr>
<tr>
<td>55</td>
<td>10.84%</td>
<td>10.80%</td>
<td>10.88%</td>
<td>3.98%</td>
<td>3.87%</td>
<td>4.09%</td>
</tr>
<tr>
<td>70</td>
<td>11.19%</td>
<td>11.15%</td>
<td>11.23%</td>
<td>4.68%</td>
<td>4.57%</td>
<td>4.79%</td>
</tr>
<tr>
<td>85</td>
<td>11.22%</td>
<td>11.18%</td>
<td>11.26%</td>
<td>5.56%</td>
<td>5.44%</td>
<td>5.67%</td>
</tr>
<tr>
<td>100</td>
<td>11.73%</td>
<td>11.69%</td>
<td>11.77%</td>
<td>5.34%</td>
<td>5.23%</td>
<td>5.45%</td>
</tr>
</tbody>
</table>

Table 3 – Means and 95% confidence intervals for digit length

Digit length is the dimension of the digit perpendicular to the strip. It has a significant effect on accuracy, however it is very small ($F(3, 627504) = 308.705, \eta^2_p = .001$). Digit Length of 55% had the lowest, and 100% had the highest average errors. There is a proportionally much stronger interaction effect with touch-size ($F(6, 627504) = 650.756, \eta^2_p = .006$). Digit length causes the strips to overlap, effectively increasing their width. This changes the periodic effects described in the touch-size section. Figure 8 compares the digit lengths of the interdigitated sensors using the large touch size and 35% digit length. With increasing digit length, the ‘dead zones’ become increasingly smaller, until zones appear where changes can be detected two separate strips simultaneously.

Not all position estimation algorithms can fully leverage this though, which is reflected in an even stronger three way interaction between digit length, touch size and method ($F(42, 627504) = 481.955, \eta^2_p = .008$).

There is also a – comparatively – strong effect of digit length on precision ($F(3, 56304) = 159.583, \eta^2_p = .008$), post hoc tests show that all levels are different, but most notably the non-interdigitated sensor performs best.

**Digit Width**

*Digit width* is the dimension of the digit parallel to the strip and had the strongest effects on accuracy and precision of all physical parameters. Reducing the digit width has a positive effect on accuracy ($F(3, 627504) = 564.239, \eta^2_p = .003$) with all levels showing a significant difference to each other, except for the non-interdigitated sensor and 75% condition. Reducing digit width has a strong positive effect on precision ($F(3, 56304) = 848.321, \eta^2_p = .043$). Large touch sizes seem to benefit most from this, which is also demonstrated by an interaction effect with touch size ($F(6, 56304) = 249.334, \eta^2_p = .026$). Reducing the digit width can, on average reduce the error over a non-interdigitated sensor, however at the cost of precision. Reducing the width increases precision, the trend suggests digit widths <35% will match the precision of non-interdigitated sensors (Figure 9).

**Method**

One of the strongest effect found in our analysis was that of method on accuracy ($F(7, 627504) = 31596.752, \eta^2_p = .261$). This is most likely in part driven by the inclusion of the NAIVE estimator which produced on average more than triple the error of the two best algorithms. mTOUCH and LINE performed similarly well on accuracy and were significantly better than all others. We also found no significant difference between COM and CUBIC, but beside those all methods were significantly different (Figure 10).

We found small, but significant simple interaction effects with both digit width ($F(21, 627504) = 63.642, \eta^2_p = .002$) and digit length ($F(21, 627504) = 95.596, \eta^2_p = .003$). We also found comparatively strong interaction effects with touch size ($F(14, 627504) = 281.349, \eta^2_p = .006$), and a three way interaction with touch size and pressure ($F(14, 627504) = 603.918, \eta^2_p = .013$) as well as digit length and pressure ($F(21, 627504) = 175.997, \eta^2_p = .006$). This suggests that the interpolation methods do not perform equally on all sensors, and that there is even larger variation based on varying pressure and size of the touch. In general, curve fitting methods performed best with small touch-sizes and low pressure, while the geometric methods performed best for strong pressure and large touch.

Finally, there was also an effect of method on precision ($F(7, 56304) = 98.828, \eta^2_p = .012$). Here GAUSS dethrones mTOUCH as the best solution. LINE has become significantly worse than mTOUCH (Figure 10).

There was a significant effect of method on consistency ($F(7, 384) = 15.257, \eta^2_p = .417$). Post hoc tests showed that NAIVE performed significantly worse than all others. CUBIC performed better than NAIVE but was also significantly worse than the remaining methods. There was also a significant interaction effect of method*touch size ($F(14, 384) = 11.364, \eta^2_p = .293$). This effect is, however, completely due to the NAIVE condition, which performed best with small and worst with large touch sizes.

**DISCUSSION**

Our analysis demonstrates that triangular interdigitation can improve the accuracy of eTextile pressure sensor matrices. We found that for all digit lengths, the accuracy is improved on average, but at the cost of precision. We found that wide digits performed worse than non-interdigitated sensors. The 55% digit width condition, however, already outperformed the non-interdigitated sensor in terms of accuracy and the 35% digit width condition was even significantly more accurate than both 55% and non-interdigitated, while approaching the non-interdigitated sensor in terms of...
For the large touch size, the interdigitated designs outperformed the non-interdigitated designs at all levels of digit width and digit height for both 

**accuracy** and **precision**.

Regarding peak estimation method, we found that both the mTOUCH and the LINE algorithm performed significantly better than all others on 

**accuracy**. We speculate, however, that they might feel very differently. We believe that there is a certain subjective dimension to interpolation which should not be ignored. We draw an analogy to the effects of latency on musical input – if latency is consistent, it is perceived as a property of the feel of the instrument, but if the latency varies, it is perceived as negatively impacting the quality of the instrument [12]. In our case it is not clear how relevant the average accuracy or even precision is, compared to the consistency and ‘feel’ of the errors. The LINE method, for example, has a distribution of errors which appears chaotic and somewhat difficult to predict compared to the mTOUCH method. The GAUSS method, on the other hand has relatively larger errors, but very good **precision** and **consistency** over pressure levels. The CUBIC method appears as a reasonable choice in terms of **accuracy**, however it performed significantly worse than all others in terms of **precision**. A visualization comparing interpolation methods for the non-interdigitated sensor and the 35% digit width, 100% digit height condition can be seen in Figure 11. The supplementary material includes these for all sensors.

While the mTOUCH method performed well on the clean data, anecdotally, we observed that the mTOUCH method led to periodic jumps when testing on the non-normalized dataset. In the normalized dataset this did not happen, presumably due to removing the noise-floor. The COM method did not display this problem.

An Excel sheet for exploring all combinations is attached in the supplementary material as well as our GitHub repository.

**Recommendations**

Overall, our data suggests using a digit length of 85%, digit width of 35% or smaller, touch size 150% (or strip size 66% of touch size) and either mTOUCH or LINE interpolation method. If the strip is wider than the touch size, then PARA is the preferred algorithm. Of all combinations tested, the combination which minimized errors and variance was 100% digit length, with 35% digit width, 150% touch size and using the mTOUCH method (see also Figure 11, right).

We suggest method and electrode design should match the application: For example, if one intends to build a footstep sensor for children, one might start with the assumption that a child’s foot is 6cm wide and space electrodes 4cm apart (making the touch size 150%). To maximize **accuracy**, one might design the electrodes to minimize digit width and use 100% digit height and use mTOUCH method.

If one is building a music controller and values **precision** over accuracy, one might choose the GAUSS method to ensure consistent behavior. Assuming a fingertip is 15mm wide, the strip spacing should be ~10mm. As interdigitation reduces precision, one might choose not to use interdigitation and instead map the electrodes so the periodic errors can be easily understood – for example one strip per half tone. If one does not wish to sacrifice accuracy, one could consider adding small digits, e.g. 55% digit width while minimizing digit width. Additional resources for parameter choice can be found in the supplement and our GitHub repository.

**Comparisons with the state of the art**

Comparing the input fidelity of matrix pressure sensors is difficult, as details are often not reported, or design parameters and goals differ greatly. For example, RESi demonstrates a potentially high electrode density [21] and
Donneaud’s and Rosenberg’s designs. Donneaud et al. [6], reporting an error of 4.89% and 6% respectively. Our best result (35% digit width, 100% digit height, COM) almost doubles the accuracy of the eTextile Matrix by Donneaud with 3.09% and even improves on the relative error size of Rosenberg’s unMousePad. The more consistent result achieved with the mTOUCH approach with the same physical sensor but averaging both gentle and strong pressure has a slightly larger average error at 3.875%, but still outperforms both Donnaud’s and Rosenberg’s designs.

Limitations and Future Work

The sensors we designed are flat, while many applications of fabric pressure matrices are not. Especially wearable solutions need to conform to the shape and movements of the body [9] which inevitably leads to artefacts and noise [10]. While any sensor design would be subject to such problems, and we therefore believe that our design recommendations still hold, it is possible that some algorithms perform better than others when subject to noise. Hybrid sensing has been suggested as a way to minimize such problems [31].

We maximized the pressure sensitive area of our sensors. Other approaches, such as sensors using zebra-fabric [22] or pressure sensitive threads [21], have zones that are not sensitive to pressure between electrodes. It is not clear how the tested algorithms perform under such conditions. While we encourage that such designs utilize interpolation, our work is aimed specifically at interdigitated sensors.

Our work points towards periodic interactions between the ratio of touch size and strip width and that of digit lengths (Figures 7 and 8). Our data and the analysis by Donneaud et al. [6] suggest that this is a topic that deserves further exploration, for example by disrupting the periodic interactions through varying strip-widths. We created and share a prototype tool for testing electrode patterns which could be used to explore this. Additional work is however required to validate this tool using the dataset we provide.

Fisher and Naidu [7] suggest reducing error by modelling the periodic effects. This would also be relevant to eTextile pressure matrices. All peak estimation methods tested by us assumed an underlying behavior of the signal distribution. As we do not believe any of the estimators we tested to be ideal, we suggest that future work either model the signal distribution of the piezo resistive material, or use parameter free approaches, such as Gaussian Processes [39]. Finally, to fully utilize interdigitation, we need better prototyping methods such as screen printing, etching, or in-situ polymerization.

CONCLUSION

This paper presents an exploration of factors influencing the design of eTextile sensor matrices. Using a combination of 100% digit length, 35% digit width, 150% touch size and the mTOUCH method we are able to achieve a relative sensing accuracy higher than reported for previous pressure sensor matrices, both textile and non-textile.

While we are pleased with this result, our work is primarily intended as a resource for designers of textile input devices. We therefore share our data-set for testing algorithms under various conditions, we provide an evaluation showing the importance of digit width, touch size and algorithm choice, and we share implementations of our algorithms as Processing functions. The supplementary material of this paper also contains visualizations of the errors and variability of errors for all sensors, all touch sizes and all pressure levels. All code, data and additional material can also be found at https://datapaullette.github.io/interdigitation/

Our investigation shows that currently the resolution at which the digits can be manufactures in textile limits the utility of interdigitated pressure matrix designs, however, our work also suggests that the approach is promising and that it is worthwhile investigating new methods for prototyping at high resolution in fabric.

ACKNOWLEDGEMENTS

This work was supported by the European Research Council, grant no. 648785, Labex SMART (ANR-11- LABX-65) and French state funds under reference ANR-11-IDEX-0004-02.
RELATED WORK


[38] HOW TO GET WHAT YOU WANT. Retrieved August 7, 2018 from http://www.kobakant.at/DIY/?p=6607
