

MechSense: A Design and Fabrication Pipeline for Integrating Rotary Encoders into 3D Printed Mechanisms

ANONYMOUS AUTHOR(S)

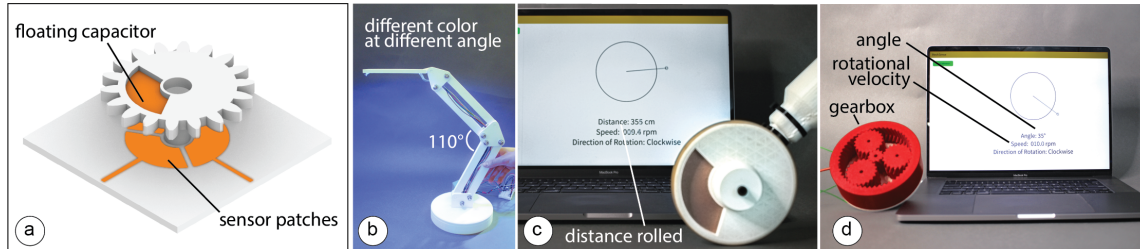


Fig. 1. (a) MechSense allows 3D printed mechanisms to sense their direction of rotation, speed, and angular position using a floating capacitor on the moving part and sensor patches on the static part of the mechanism. (b) A smart linkage-based lamp that changes the color of the light when the linkage bars are moved. (c) a distance measuring wheel that can measure the dimension of surfaces, and (d) a planetary gear box that can sense its own state.

We introduce *MechSense*, 3D-printed rotary encoders that can be fabricated in one pass alongside rotational mechanisms, and report on their angular position, direction of rotation, and speed. *MechSense* encoders utilize capacitive sensing by integrating a floating capacitor into the rotating element and three capacitance sensor patches in the stationary part of the mechanism. Unlike existing rotary encoders, *MechSense* does not require manual assembly but can be seamlessly integrated during design and fabrication. Our *MechSense* editor allows users to integrate the encoder with a rotating mechanism and exports files for 3D-printing. We contribute a sensor topology and a computational model that can compensate for print deviations. Our technical evaluation shows that *MechSense* can detect the angular position (mean error: 1.4°) across multiple prints and rotations, different spacing between sensor patches, and different sizes of sensors. We demonstrate *MechSense* through three application examples on 3D-printed tools, tangible UIs, and gearboxes.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**.

Additional Key Words and Phrases: 3D printed mechanisms, printed electronics, capacitive sensing.

ACM Reference Format:

Anonymous Author(s). 2022. MechSense: A Design and Fabrication Pipeline for Integrating Rotary Encoders into 3D Printed Mechanisms. 1, 1 (September 2022), 28 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

Advances in 3D printing over the last decades have enabled increasingly complex 3D printed objects, including objects with moving parts that contain mechanisms, such as gears, linkages, and wheels (Grafter [15]). While early 3D printed

Permission to make digital or hard copies of all or part 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 components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2022 Association for Computing Machinery.

Manuscript submitted to ACM

Manuscript submitted to ACM

mechanisms were fully passive, researchers have started exploring how to augment 3D printed mechanisms with sensors to enable interactive applications.

To sense the motion of these 3D printed mechanisms, researchers traditionally used external sensors. For instance, researchers have used acoustic sensing via external microphones (*Lamello* [16]) or used hall effect sensors and magnets (*MechaMagnets* [23]) to determine the interaction with 3D printed objects. Using external sensors, however, requires additional assembly.

Recently, the advent of multi-material 3D printing with conductive filament has enabled the integration of sensors with 3D printed geometries. For instance, researchers have shown how to print capacitive sensors from conductive filament to integrate touch sensors (*Capricate* [18]) and deformation sensors (*MetaSense* [9]) with 3D printed object geometries. Thus, a variety of different sensors can now be 3D printed together with the object geometry in one pass, facilitating the creation of interactive objects.

For rotational mechanisms, such as gears, linkages, and wheels, however, no sensor design has been proposed yet that can be 3D printed in one pass. To sense direction of rotation, speed, and angular position of rotational mechanisms, existing encoder designs require additional manual assembly. For instance, *SteelSense* [21] enables high-resolution sensing of rotational elements, such as hinges and ball bearings, but requires metal casting of the sensing elements that afterwards have to be manually integrated into a 3D printed casing. Similarly, Karali et al. [10] demonstrate a capacitive rotation sensor that utilizes two patterned copper plates that change their capacitance depending on the relative angle to each other. However, the copper plates cannot be 3D printed together with the object geometry, and thus, require manual assembly.

In this paper, we introduce *MechSense*, a fabrication pipeline based on conductive multi-material 3D printing that can print the geometry of the mechanisms and sensors together in one pass without the need to assemble the conductive and non-conductive parts. We focus on rotating mechanisms and develop a sensor layout that utilizes tracking of a floating capacitor which can augment various rotational mechanisms with the ability to sense their direction of rotation, speed, and angular position (Figure 1). To better understand user’s familiarity with multi-material 3D-printing and to identify the design software they most commonly use, we conducted a survey of 20 hardware design professionals. Based on the survey results, we built a 3D editor add-on for SolidWorks that automatically integrates the sensor layout into the mechanism’s geometry and exports files for 3D printing. In our technical evaluation, we determine the angular position estimation error for sensors with different spacing between sensors, the effect of the sensor patch size, and the impact of the proximity of the user’s hand, near the *MechSense* encoder. We demonstrate the usefulness of this method with three application examples for tangible user interfaces, construction tools, and gearboxes.

In summary, we contribute:

- a *sensor layout* based on a floating capacitor that can be integrated with 3D printable rotational mechanisms to determine their direction of rotation, speed, and angular position (mean error: 1.4°);
- a *3D editor extension* that automatically integrates sensors into mechanical components and generates the files for 3D printing;
- a *Java/Processing* tool that converts the raw sensor data into angular position, direction of rotation, and speed.
- a *technical evaluation* of the angular position accuracy for different spacing between sensors, the effect of the size of the sensor patch, and the influence of a user’s hand in proximity to the capacitive sensor;
- three *applications* that demonstrate sensing integrated with various rotational mechanisms for tangible user interfaces, construction tools, and gearboxes.

2 RELATED WORK

Our work is related to research that investigates how to fabricate rotary encoders, 3D print sensors, and integrate sensors into mechanical elements.

2.1 Rotary Encoders Using Capacitive Sensing

Many rotational mechanisms use encoders to retrieve data on the position and the speed of the rotating element. Most commercial encoders utilize optical or inductive sensing approaches, but researchers have also investigated ways to fabricate capacitive encoders since the sensor elements do not require contact between the stationary sensor and the rotating element. For example, Cermak et al. [3], Gasulla et al. [4], Ferrari et al. [2], and Karali et al. [10] developed rotational encoders that consist of two stationary circular conductive plates opposite of one another, where one of the plates is segmented into electrodes that acts as a capacitive pair with the other stationary plate. The two stationary plates are separated by a rotating insulator or a conductive plate that triggers changes in the capacitance for each electrode. In all these approaches, the suggested encoder geometry consists of at least three plates that have to be individually fabricated, for example, by copper etching, and have to be manually mounted on a motor shaft. Additionally, Zheng et al. [24], Hou et al. [7] and Wang et al. [22] demonstrated approaches that utilize a pair of conductive circular plates, with one being a stator and one being a floating conductive rotor that is segmented into two parts via a complex pattern that trigger changes in capacitance when the rotor plate is in motion. All of these methods require manual assembly of the capacitive copper plates into a (3D printed) mechanism. In contrast, MechSense enables users to print the entire object with integrated mechanisms and rotational sensors in one pass.

2.2 3D Printed Sensors

More recently, researchers started to use 3D printing with conductive filament to fabricate objects with integrated sensors in one go. One example of this is the integration of direct touch sensing with 3D printed objects. *Capricate* [18] and *PrintPut* [1], for instance, provide editors to embed touch sensors, sliders and touch pads into 3D printable objects. Similarly, *Let's Frets!* [14] is a 3D printed capacitive fretboard that detects the user's fingers to help teach guitar playing. In addition to 3D printing the sensor geometry, *ModElec* [6] also automatically creates the internal conductive circuit traces necessary to route the sensor to an external microcontroller.

Rather than using 3D printing with conductive filament for direct touch sensing, researchers also investigated how to embed sensors that interact with other objects, such as capacitive surfaces. For instance, *Flexibles* [20] are tangible objects with embedded conductive material whose deformation can be sensed via a capacitive touch screen. *CAPath* [11] extends this work by also providing tangible sliders and knobs. *itsy-bits* [19] are 3D printed tangibles with embedded conductive markers that can be used to identify which tangible is used on a capacitive screen. Similarly, *3D-Auth* [13] are 3D printed tangibles with conductive patterns, which allow for two-factor authentication. *Off-line sensing* [17] 3D prints conductive sensors inside 3D objects connected to channels that contain liquids. When the object moves, the liquid bridges the gap between the capacitive areas, which can be used to detect orientation. However, none of these works investigate how to use conductive 3D printing to embed sensing into mechanical elements.

2.3 Integrating Sensing into Mechanical Elements

Over the last decade, the majority of work that added sensing to 3D printed mechanical elements used external sensors. For example, *Lamello* [16] uses acoustic sensing to detect interaction with physical sliders and rotary knobs, i.e. when the

157 mechanism gets rotated or moved, it makes a noise which can be sensed with a microphone. *MechaMagnets* [23] integrates
158 hall effect sensors and magnets into 3D printed objects for haptic and physical motion feedback. *StrutModeling* [12]
159 uses rotational encoders in the metal connectors of 3D printed strut elements to sense how users assemble the struts.
160 *3D Printing Wireless Connected Objects* [8] 3D prints gears and springs and adds conductive copper tape after printing
161 has finished to create wifi signals when the mechanisms move.

163 More closely related to our work is research that uses conductive multi-material 3D printing to integrate sensing
164 into mechanical structure. *MetaSense* [9] uses conductive 3D printing to integrate shear sensing into mechanical
165 metamaterials. *FlexKeys* [5] similarly uses conductive multi-material 3D printing to create deformable springs that
166 require no support material and can be used as sensors integrated with input devices such as keyboards. However,
167 these works do not show how to integrate sensing with rotating mechanical elements, such as gears and linkages.
168 While *SteelSense* [21] focuses on rotating mechanical elements, such as gears, hinges, screws, and bearings, it requires
169 assembly since the conductive parts are metal-cast separately from the rest of the 3D print.
170

172 In summary, the existing work either requires multiple fabrication steps and manual assembly, or does not support
173 rotating mechanisms, such as gears, linkages, and wheels. In contrast, *MechSense* provides an end-to-end fabrication
174 pipeline for integrating *MechSense* encoders into 3D printable mechanism, provides a sensor layout that generalizes
175 across different rotational mechanisms, and contributes a computational model to convert raw sensor data into angular
176 position, direction of rotation, and rotational speed.
177

179 3 SURVEY OF HARDWARE DESIGN PROFESSIONALS

181 We conducted a survey of product design engineering professionals at a large cross-industry consulting company to
182 understand their familiarity with multi-material 3D printing, other smart materials, and the design engineering process
183 in general. In doing so, we hoped to identify features which may make our design tool more effective, particularly in
184 regard to existing utilized tools. 20 professionals responded to the survey including mechanical engineers (7), electrical
185 engineers (7), industrial designers (5), and a product manager (1).
186

188 One key insight was the prevalence of participants who stated they had challenges integrating mechanical and
189 electrical systems (50%). P17 explains 'mechanical constraints are sometimes in conflict with electrical requirements'
190 and P13 goes on to say 'electronic component placement vs. mechanical volumetric constrains [are often a challenge].'
191 Additionally, 14 participants shared interest in using or have already used advanced materials and manufacturing
192 methods, such as multi-material 3D printing, with 18 being interested in creating smart products with these materials
193 and methods in the future. Broadly, such responses indicated a need for better integration of mechanical and electrical
194 components, and an opportunity to accelerate and simplify prototyping in electro-mechanical products.
195

196 In terms of existing design tools used by our respondents, we found SolidWorks is the most commonly used CAD
197 modelling software (n=10), followed by Rhino (n=5) and Fusion360 (n=2). Based on these results, we created an add-on
198 for SolidWorks that allows designers to automatically integrate the sensing directly into the design of the mechanical
199 components for faster and easier prototyping.
200

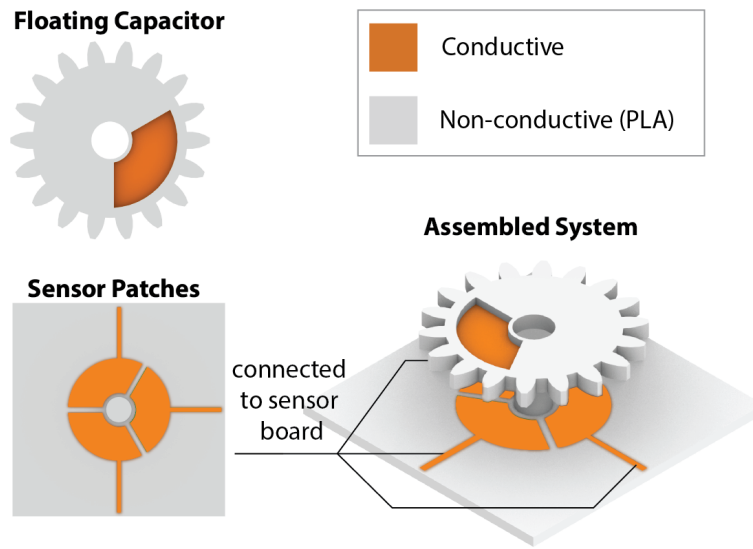
202 4 MECHSENSE

204 MechSense is a method that uses multi-material 3D printing with conductive filament to augment 3D printed rotational
205 mechanisms, such as gears, linkages, and shafts, with sensing capabilities. It uses capacitive sensing to detect the speed
206 of rotation, direction of rotation, and angular position.
207

209 **4.1 Sensor Design**

210 When deciding on how to design the sensor, we took into account that rotational mechanisms all move around a static
 211 shaft. The obvious approach would be to add an active capacitive sensing patch to both the shaft and to the rotating
 212 component, such as a gear, where both components are connected to sensing circuitry. However, the sensor on the
 213 rotating element has to be in constant connection to the 3D printed circuitry, which is difficult to achieve since 3D
 214 printed electrodes might frequently loose connection or have a high and variable resistance.
 215
 216

217 Therefore, we developed a capacitive sensor design that incorporates a floating capacitor into the rotating component,
 218 such as the gear, and three sensor patches arranged in a circle around the static component, such as a shaft or a base
 219 plate (Figure 2) that can be fully 3D printed in one pass. Using capacitive sensing with this geometry eliminates the
 220 need for direct contact to the sensor patch in the moving component of the mechanism, thus allowing the moving
 221 component to rotate freely, providing a consistent sensor signal.
 222
 223



244 Fig. 2. Our sensor layout consists of a floating capacitor in the moving part of the mechanism and three sensor patches on the static
 245 part of the mechanism that are wired to a capacitance sensing board.
 246

247 The sensor patches on static part of the mechanism are connected to a capacitance sensing board (FDC2214 Sensor
 248 Board). The floating capacitor is integrated into the moving part of the mechanism and is not wired to the circuitry.
 249 Instead the floating capacitor creates a coupled capacitance system between neighboring sensing patches. The amount
 250 of overlap between the floating capacitor and the sensor patches impacts the capacitance of the system, which we
 251 utilize to determine the location of the floating capacitor. This enables us to track the moving part of the mechanism
 252 while it is freely rotating.
 253
 254
 255

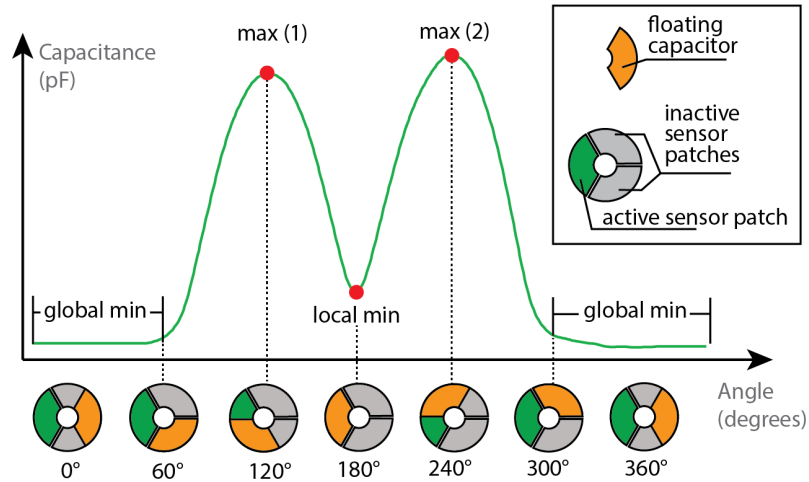
256 **4.2 Sensor Signal Properties**

257 Figure 3 shows the signal of one sensor patch while the floating capacitor is rotating by 360°. We observe four different
 258 features in the signal: (1) a global minimum, (2+3) two maxima, and (4) a local minimum.
 259
 260

261 **Global Minimum (No Overlap):** When the floating capacitor and the active sensor patch have no overlap, we observe
 262 a *global minimum* in the signal (Figure 3 at 300° - 60°). The reason for this is that the floating capacitor is not coupled
 263 with the sensor patch and thus the detected capacitance is lower than in the other states.
 264

265 **Maxima #1 and #2 (Half Overlap):** When the floating capacitor half overlaps with the active sensor patch and one
 266 of the neighboring patches, we observe two maxima. Although the shape of the sensing patches is identical, differences
 267 in print quality create different conductivity across patches. This results in different peaks for each sensor rather than
 268 identical global maxima for all sensor patches. We observe these maximum values because the floating capacitor has an
 269 even overlap between the sensing capacitor plate and the neighboring capacitor plate which acts similar to a grounded
 270 patch as it is connected to the sensing board. Thus, we observe a coupled capacitance that is maximal if the floating
 271 capacitor has a half overlap with both the active sensing patch and one of the neighboring patches (Figure 3 at 120° and
 272 240°).
 273
 274

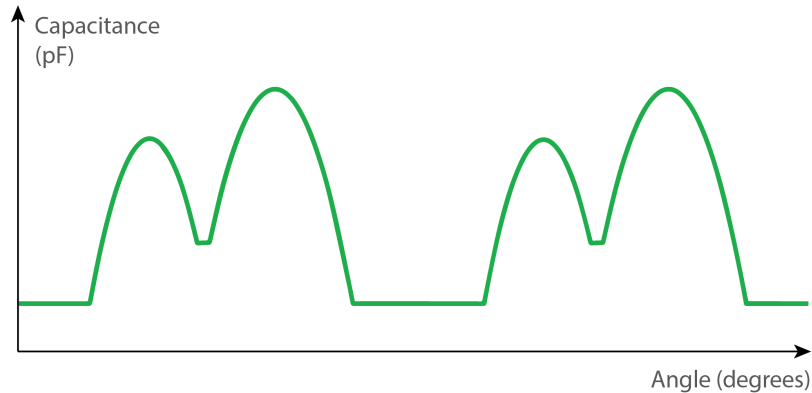
275 **Local Minimum (Full Overlap):** When the floating capacitor and the active sensor patch are aligned, we observe a
 276 *local minimum*. The reason for this is that the floating capacitor has no overlap with the neighboring patches but is
 277 still in close proximity to them which leads to a small but measurable coupled capacitance (Figure 3 at 180°). Thus, the
 278 measured sensing value is still larger than the global minimum since the floating capacitor has no overlap with the
 279 sensing patch (Figure 3 at 360°).
 280
 281



282
 283
 284
 285
 286
 287
 288
 289
 290
 291
 292
 293
 294
 295
 296
 297
 298
 299
 300 Fig. 3. Signal profile of one sensor patch when the floating capacitor overlaps by different amounts and corresponding angular
 301 positions. We observe maxima and minima of the sensor signal at distinct positions of the floating capacitor.
 302

303 **Simulation:** To better predict and explain the theoretical response of a particular sensor geometry, we built a com-
 304 putational simulator in Python that predicts the waveform by calculating the capacitance over the position for each
 305 sensing element. It then combines these into a model of the overall system capacitance. The model takes as input the
 306 following parameters that describe the sensor geometry: inner and outer radii of sensing patches integrated into the
 307 rotating component, the distance between the sensing plates and passive elements, the size of the sensing plates minus
 308 the gap between plates, and the width of the passive element. First, the capacitance of a single sensing element is
 309 modelled independently as a function of the floating element's position. Then the independently modelled sensors are
 310
 311

313 combined into a model of system capacitance by treating the primary sensing patch as a capacitor in series with all
314 other capacitors, which are connected in parallel to an arbitrary, yet stable, reference. Figure 4 shows our simulation
315 result, which exhibits a similar waveform as the measured sensor signal.
316



317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332 Fig. 4. Simulation of the signal profile of one sensor patch for two full rotations.

333 334 335 4.3 Sensing Angular Position

336 Our goal is to derive the angular position from the capacitance values that we read from each of the sensor patches
337 while the floating capacitor is rotating over them. To do that, we proceed with the following steps: (1) We preprocess
338 the sensor data with a low-pass filter to reduce noise; (2) We determine the capacitance values of each extremum in
339 the signal (e.g. the local minimum); (3) We divide the signal into 5 segments with 4 segments located between two
340 extrema and one segment in the constant global minimum (Figure 5a); (4) We normalize the segments that are not
341 in the constant global minimum and fit a polynomial to the sensor values that approximates the signal within each
342 segment (Figure 5b). We can compute the angular position of the rotating element by determining the current signal
343 segment it is in and estimating the current angle using the polynomial of this segment; and (5) we estimate the angle
344 across all 3 sensor patches and calculate the average of all estimates. Since the sensor signal is constant in the global
345 minimum across 120° (i.e. the floating capacitor has no overlap with the sensor patch), we cannot derive any angle in
346 this region. Thus, we designed the sensor to contain 3 sensor patches such that the floating capacitor in the moving
347 element always overlaps with at least one sensor patch at all times. If more than one sensor patch is not in the global
348 minimum and outputs an angle estimate, we average between multiple sensor patch estimates.
349
350
351
352

353
354 **Preprocessing of Raw Sensor Data:** The raw sensor data contains noise that leads to the occurrence of multiple false
355 local extrema. Hence, we apply a moving average filter to smooth the data. We chose a window size of 8 that is large
356 enough to obtain a smooth signal but is also small enough to preserve the magnitude of the extrema.
357

358 **Detecting Extrema in the Sensor Signal:** The key features for our angle estimation algorithm are the minima and
359 maxima in the sensor signal over a full rotation. We find these extrema during an initial calibration step which has
360 to be done once before using the sensor (Section 4.4). In this calibration step, we fully rotate the mechanism 3 times
361 and use a peak detection method to identify the extrema. After storing the sensor values for these extrema, we use
362 them for detecting extrema in live data by thresholding all incoming sensor values for these extrema points. We also
363
364

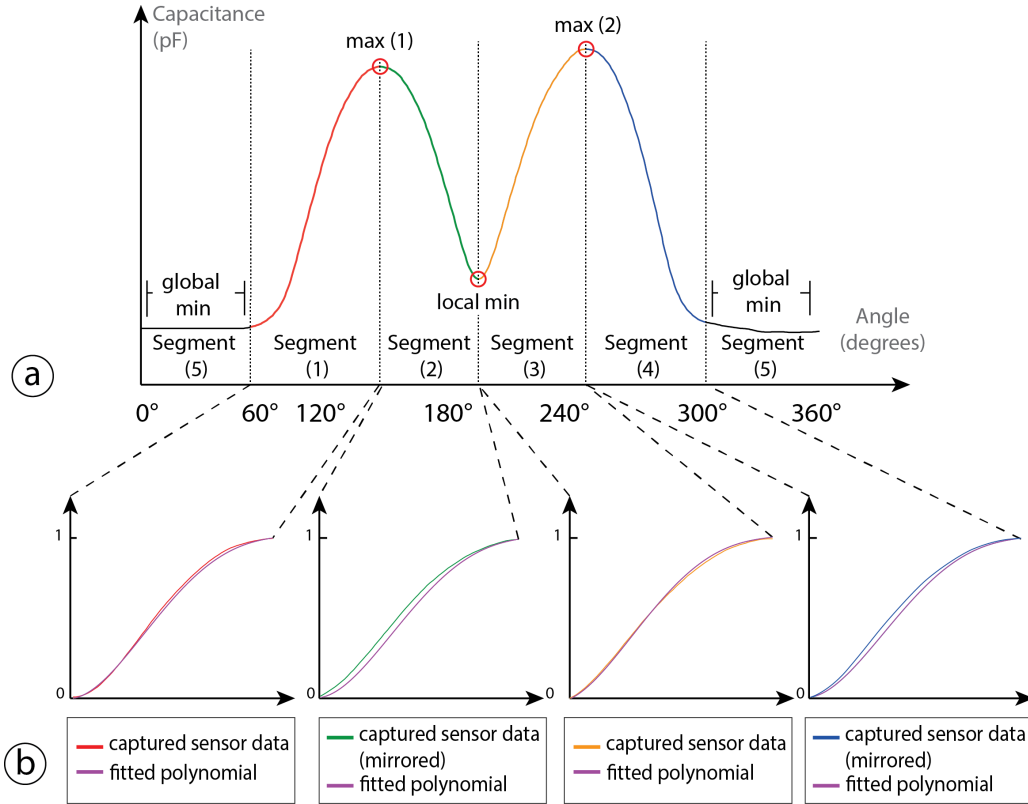


Fig. 5. Processing of sensor data to estimate the angular position of the rotating element. (a) Segmenting the sensor signal into 5 segments where each segment corresponds to a 60° rotation of the element with the exception of segment 5 which spans over 120° . Each segment is bijective, i.e. there is only one possible angle for a reading from the sensor. (b) Approximating segment 1-4 with a 4th-degree polynomial. We invert segment 2 and 4 such that all segments are monotonically increasing.

take into account that the rotating element cannot randomly jump from one angular position to another, i.e. if the rotating element just passed through the local minimum, the next possible extremum can only be the maximum #2 if rotating clockwise (Figure 5a). Thus, we keep track of our last visited extrema and threshold only for the next possible extremum.

Segmenting the Sensor Signal: We divide the sensor signal into 5 segments where 4 segments are between two neighboring extrema of the signal and the fifth segment lies within the global minimum of the sensor (Figure 5a). For example, segment 1 is between the global minimum and the first maximum which corresponds to when the floating capacitor is between 60° and 120° . Within each segment, the sensor signal is strictly monotonically increasing or decreasing which means that for a specific sensing values there is only one possible angle position. We segment the sensor signal because one sensing value can have multiple possible angle positions throughout the entire signal.

Fitting a polynomial to each segment: Our goal is to generate a 4th-degree polynomial that approximates all sensing values per segment and outputs an angular position for a sensor reading. To do that, we first ran an experiment by rotating a gear with an integrated MechSense encoder 50 times and captured the generated sensor data (3mm patch

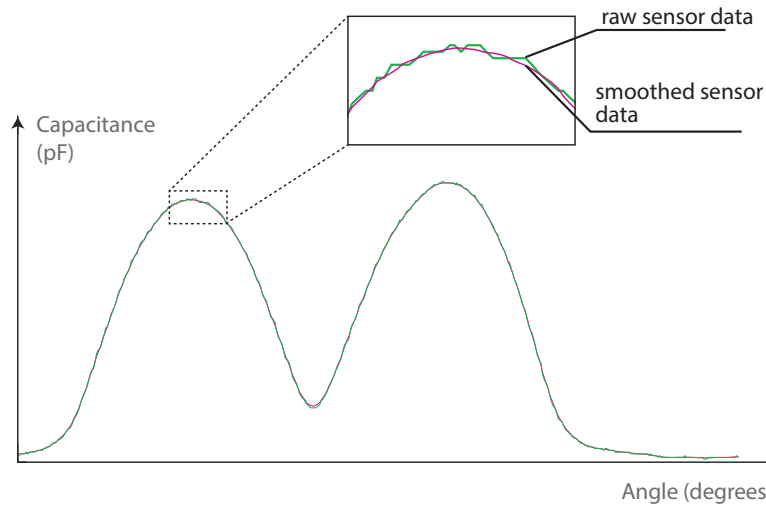


Fig. 6. Smoothing sensor data. We apply a moving average filter (windows size = 8) to the raw sensor data to attenuate local fluctuations.

distance, 765mm^2 sensor patch size, 10 RPM, Figure 18). We then segmented the sensor data as described above. Next, we normalized all sensor data within each segment using the extrema values and mirrored the data for segment 2 and 4 such that each segment showed only increasing sensing values (Figure 5b). After that we fitted a 4th degree polynomial to each of the 4 segments (that are not the constant global minimum), i.e. we generated 200 polynomials (50 rotations x 4 segments). Finally, we averaged the coefficients across all 200 polynomials. The resulting polynomial of the form $f(x) = ax^4 + bx^3 + cx^2 + dx + e$ has the coefficients $a = 1.74 * 10^{-7}$, $b = -2.86 * 10^{-5}$, $c = 0.00131$, $d = 0.0034$, and $e = -0.00863$.

After defining our approximation polynomial, we can now apply the polynomial directly to the normalized live sensing values of each segment which generates our angle estimation. We use the same polynomial for all angular position estimations across multiple prints and for all segments. Different sensor values at the extrema are compensated through the normalization step, i.e. the magnitude of the minima and maxima might be different for an individual print which gets scaled to a range between 0 and 1 through the normalization step.

Detecting the current segment for Real-Time Sensing: When starting the sensor for the first time, the rotating element might be in an unknown angular position. To detect the current position of the rotating element, we have to detect two extrema to identify in which segment the rotating element is located. While detecting only one extrema already allows us to detect the current position of the rotating element, it is not possible to estimate the direction of rotation and the current segment in which the floating capacitor is located. For example, if the rotating element is detected at the local minimum and the subsequent sensor values are increasing, it is not clear if the rotation is clockwise or counterclockwise. Thus, the mechanism has to pass through a second extremum to identify the direction of rotation and the current segment. For example, if the rotating element passes through the local minimum and then through the global maximum, we can determine that the rotation is clockwise and the element is in segment 4.

In our implementation, we store the current location of the rotating element even when the sensor is unplugged. This allows us to keep track of the angular position also without an initial rotation through 2 extrema in case the floating capacitor has not been moved. In any other case, our algorithm can detect the angular position and the direction of rotation after passing through 2 extrema.

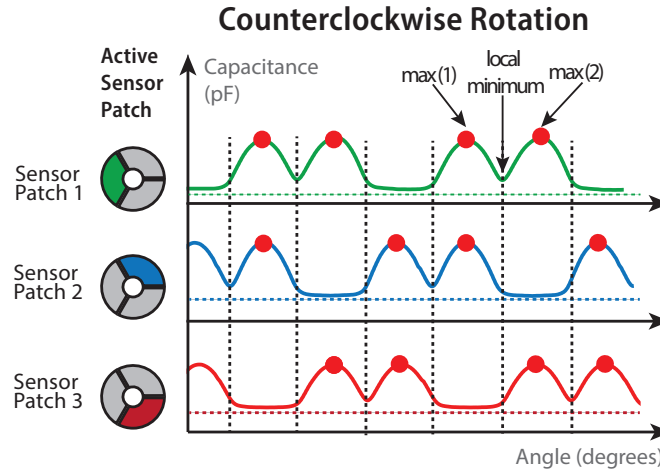


Fig. 7. Detecting extrema during calibration using sensor data of multiple patches. We first detect the global minima in all sensor patches using the 20% lowest percentile of sensor values. Next, we can use the location of the global minima to detect the maxima in adjacent patches. For example, if sensor patch 3 is in the local minimum, we can identify maximum #1 of sensor patch 1 by finding the largest sensor value of all within the range of global minimum of patch 3. Finally, we can detect the local minimum as the smallest value between two maxima.

4.4 Calibration Phase

Before our computational model can estimate the angular position of the rotating element, we need to perform a calibration in which we detect the sensing values at the extrema of the sensor signal, i.e. the local/global minima/maxima. To do that, we rotate the printed mechanism three times by 360° to collect sensor values for each feature (3 rotations \times 4 features = 12 extrema values total). We detect the extrema by taking into account all 3 sensor signals of the patches. The direction of rotation is critical during the calibration step to detect the maxima in the right order. Thus, we require the calibration to be done only rotating clockwise or counterclockwise. Our software tool allows users to set the direction of rotation before starting the calibration.

Global Minimum: We detect all sensor values that fall into the global minimum by finding the smallest and the largest sensing value throughout all rotations and define all values to be in the global minimum if they are within the 20% smallest values in the full range of values that we capture during calibration, i.e., we define all values to be a global minimum if they fall into the interval: $[\text{smallest value}; 0.2 \cdot (\text{largest value} - \text{smallest value})]$.

Maximum #1: To find the value for the first maximum, we take into account the location of the global minima of the neighboring sensor patches. If we overlay the sensor signals of all 3 sensor patches, we can see that the local minimum

of sensor patch 1 can only exist if the sensor values of patch 3 are within the global minimum (Figure 7). Thus, we detect the first maximum as the largest value that we read for sensor patch 1 while sensor patch 3 is in the global minimum. With the same strategy, we can detect the first maximum for patch 2 if patch 1 is in the global minimum, and for patch 3 if patch 2 is in the global minimum.

Maximum #2: Similar to detecting the first maximum, we can detect the second maximum as the largest value within a specific range that is defined by the neighboring sensor patches. Since the second maximum of sensor patch 1 can only exist if the sensor value of sensor patch 2 are in the global minimum, we detect the second maximum as the maximum sensor value within this range (Figure 7). Similarly, we can detect the second maximum of sensor patch 2, when the sensor value of patch 3 are in the global minimum, and the second maximum of patch 3, while sensor patch 1 is in the global minimum.

Local Minimum: To detect the local minimum we take into account the values that we found for the two maxima. Since the local minimum has to be located between these two maxima, we find the minimum value that is between these two extrema.

4.5 Compensating for Sensing Value Variations at Extrema

It is possible that the sensor values at extrema change over time. This might happen due to friction of the rotating element that might change its angle or distance to the sensor patches but also from external factors like humidity changes that influence the print material or by the proximity of other capacitive objects like the user's hand. To compensate for such variations we (1) modify the angular position estimation method to be robust against under- and overshooting of the sensor values at extrema, and (2) we update the current estimation of the sensor values at extrema throughout multiple rotations.

Robustness of Extrema Detection: To compensate for overshooting at extrema points, we clamp the sensor values, after normalization, to 1 if they are larger than 1 and to 0 if they are smaller than 0. Once the sensor values surpass 1, we detect that as an extremum and switch from one segment to the next segment. It is also possible that sensor values undershoot, i.e. they never reach a value of 1. In this case, we take into account the sensor values of the neighboring patches. If one of the patches detects an extremum (i.e. the sensor values reach 1), we switch to the next segment globally even if the other sensor patches do not detect an extremum.

Updating for Extrema variations: To compensate for changes in the extrema values over multiple rotations we continuously update the extrema value using a moving average. For example, to update the sensor value for the local minimum, we identify the smallest that we measured after passing through the local minimum. Next, we take the average between the old estimate of the local minimum and newly detected value, i.e., $LocalMin_{new} = 0.5 * (LocalMin_{old} + DetectedMin)$. We update all other extrema in the same manner.

4.6 Detecting Speed and Direction of Rotation

Speed: We compute the speed of rotation by measuring the change of angular position within a time frame. To do that, we attach a time stamp to the sensor data before sending it from the microcontroller to our implementation of the computational model and convert the change in angular position into Revolutions per Minute (RPM).

Direction of Rotation: To detect the direction of rotation, we first determine in which segment the sensor is currently. Each segment has monotonically increasing or decreasing values if rotated clockwise. For example, segment 1 only

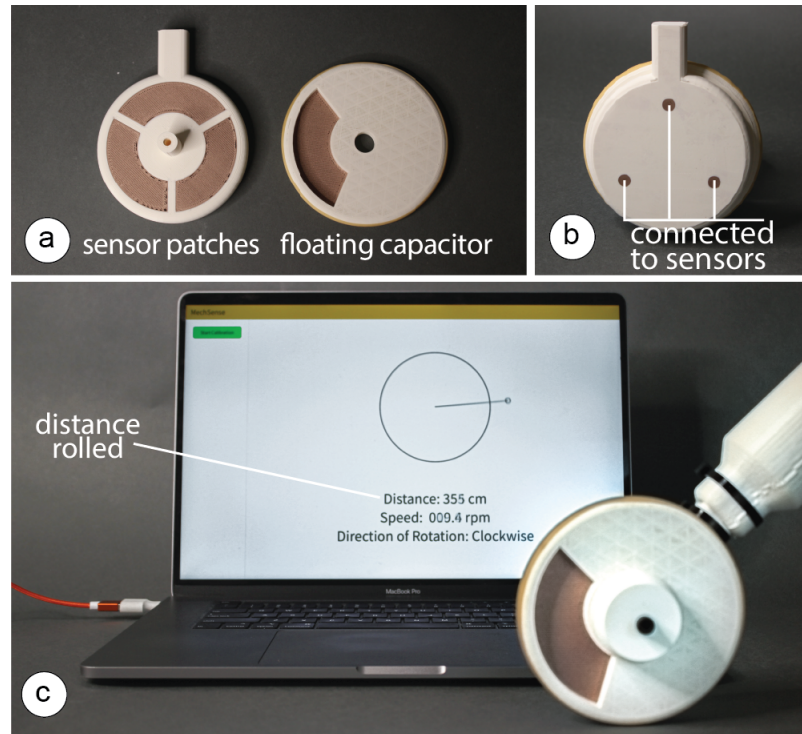
573 shows increasing values if rotated clockwise. In the case that the sensor outputs decreasing values while being in
 574 segment 1, we can identify a counterclockwise rotation. The same principle can be applied to all other segments with the
 575 exception of the global minimum which is nearly constant. Since only one sensor patch can be in the global minimum
 576 at a time, we can estimate the direction of rotation using the other two sensor patches.
 577

579 5 APPLICATIONS

580 Our technique allows us to integrate sensing with various rotating mechanisms to enable a wide range of applications,
 581 such as a distance measuring wheel, a linkage based lamp, and a planetary gear box.
 582

584 5.1 Distance Measuring Wheel

585 To show how a wheel can be augmented with sensing capabilities, we integrate sensing into a distance measuring
 586 wheel (Figure 8). A user holds the measuring wheel by the handle and then rolls the wheel over a surface to capture its
 587 dimensions. The wheel measures distance by translating rotational information into linear information. We integrated
 588 the floating capacitor into the moving wheel, and the three sensor patches into a static plate parallel to the wheel
 589 (765mm^2 patch size with 3mm separation). Once connected to the sensing board, the integrated sensors measure the
 590 angular position, which allows the user to measure the dimensions of irregular geometries and big open spaces. Since
 591 MechSense can identify the direction of motion as well, the wheel only adds to the cumulative distance when the wheel
 592
 593
 594



622 Fig. 8. A distance measuring wheel with integrated sensing that can measure perimeters of irregular geometries and large open
 623 spaces.

is moved forward and subtracts any distance when the wheel is moving backwards. Our measuring wheel has a mean error of 1.45mm as shown by our technical evaluation on the accuracy of determining angular positions (Section 7).

5.2 Smart Desk Lamp with Linkages

To demonstrate how MechSense can be used to integrate sensing into linkage-based rotation, we designed and manufactured a smart desk lamp (Figure 9), which consists of two linkage bars and a base. Rotating the linkage bars triggers different lighting conditions. Rotating the bottom linkage bar changes the color of the light, and rotating the middle linkage bar changes the brightness. We first augmented the bottom linkage bar and the corresponding static pin joint plates at the bottom of the lamp with sensors ($76mm^2$ patch size with 3mm separation). We then repeated the process for the second linkage bar and the middle pin joint plate. The sensors of each linkage bar are connected to their own sensing board.

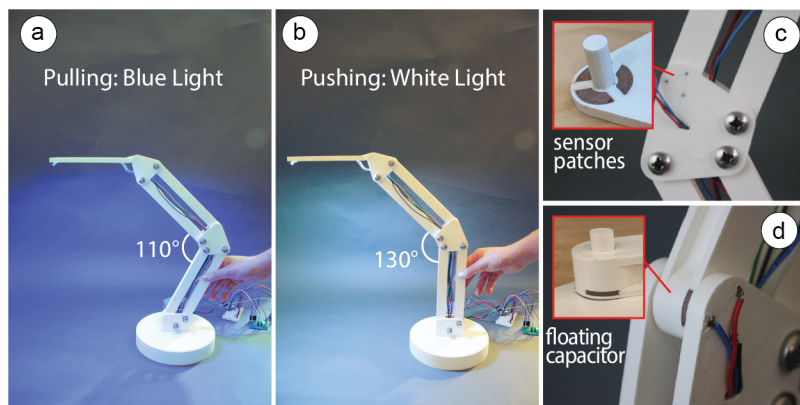


Fig. 9. The smart desk lamp allows the user to set the color of the light by rotating the bottom linkage, and the brightness of the light by rotating the top linkage. To do that, we integrated two MechSense encoders in each linkage of the lamp, respectively.

5.3 Planetary Gear Box

Integrating sensors directly into a mechanisms geometry is particularly useful in space-constrained applications.

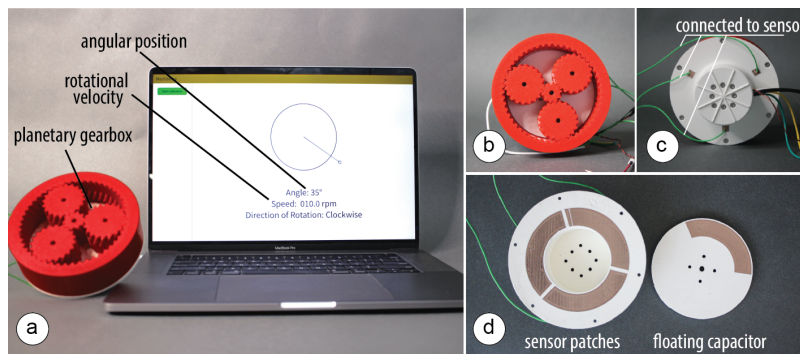


Fig. 10. This planetary gearbox can sense its own angular position and rotational velocity.

One example of this is a planetary gearbox, which integrates multiple gears into a confined volume (Figure 10). Such gearboxes can be integrated into robotic arms and connected to a motor for actuation. By printing the static part of the sensors into the motor casing and the moving part into the geometry of the planetary gears, we are able to measure the angular position and the rotational velocity without increasing the overall size of the joint or requiring extra geometry for mounting an external encoder.

6 CREATING MECHSENSE OBJECTS

Based on our survey results, we build a MechSense plugin for SolidWorks using a C# script to facilitate the creation of objects with integrated sensing. The plugin automatically integrates the sensor design into a CAD model of the mechanism and then enables exporting files for multi-material 3D printing. After 3D printing the mechanism, users need to wire the sensors to the microcontroller, and upload the sensing code which streams raw sensor data to a computer connected via a serial port. We developed an implementation of our computational model and a UI for Java/Processing. The Processing UI enables users to calibrate the sensors, as described above. After finishing the calibration, our software converts the raw sensor data from the microcontroller into the estimations for the angular position, direction of rotation, and the speed of the rotating element. The computational model is implemented as a Java library that can be imported to any user-generated UI and application.

6.1 Designing the Mechanism

We developed a plugin for Solidworks that facilitates the integration of the sensor topology into a 3D mesh.

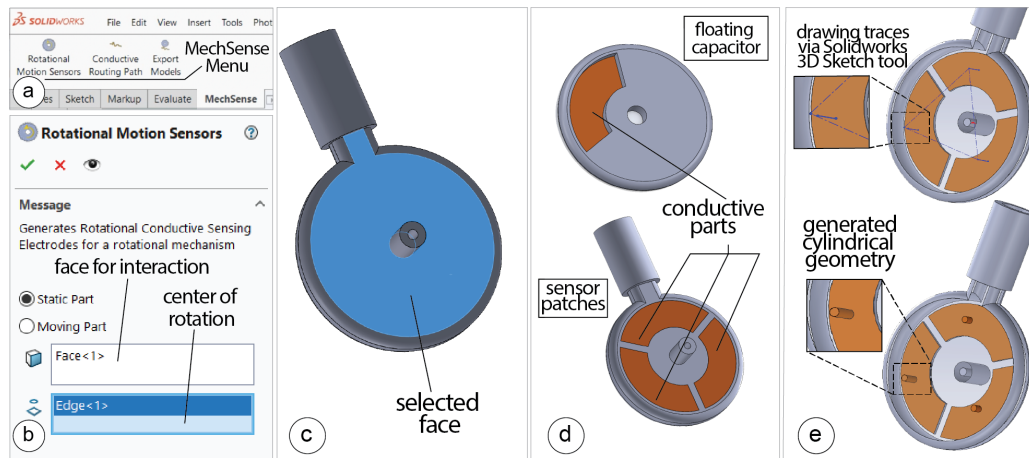


Fig. 11. Integrating sensing into a mechanism using the MechSense 3D editor plugin.

Integrating Sensor Patches into the Static Part: To integrate the sensor patches into the static part of the mechanism, the user selects the 'static part' option from the menu (Figure 11a). Next, the user clicks on the plane for the static part onto which the sensor patches should be integrated. Finally, we need to define the center of rotation around which the sensor patches will be distributed. To do that, the user selects the 'Center of Rotation' field and clicks on the circular edge of the shaft. We compute the center of rotation by finding the mean point of all vertices that are in the circular edge of the shaft. Once the face for the sensor patches and the center of rotation are defined, our plugin generates the

729 sensor layout on the static part of the mechanism. The three sensor patches are generated by defining a 3mm distance
730 between them and keeping a 1mm distance from the outlines of the static part geometry. Subsequently, our software
731 creates separate meshes for 3D printing the sensor patches with conductive filament and the rest of the geometry with
732 non-conductive filament.
733

734 **Integrating the Floating Capacitor into the Moving Part:** To integrate the floating capacitor into the moving part
735 of the mechanism, the user first selects the 'Moving Part' option from the menu bar (Figure 11) and then selects the face
736 onto which the floating capacitor should be integrated. The user next specifies the 'Center of Rotation' by clicking
737 on the circular edge of the shaft. After the user confirms their selection, MechSense integrates the floating capacitor
738 geometry with the moving part of the mechanism. It then subtracts it from the original geometry to generate separate
739 files for the conductive and non-conductive parts for multi-material 3D printing.
740

741 **Creating Traces:** Users can integrate conductive traces to connect the sensor patches to the sensing board and
742 microcontroller by using the built-in SolidWorks tools. By using the "3D Sketch" tool, the user can draw the lines for
743 the conductive traces directly onto the geometry of the mechanism. The user draws conductive traces from the sensor
744 patches to a convenient location on the mechanism to connect to the sensor board and microcontroller. Our MechSense
745 plugin converts this path into a cylindrical geometry of 2mm diameter for 3D printing upon the user selecting the
746 "Conductive Routing Path" button.
747
748
749

750 **Exporting Geometry and Sensing Code:** On export, the MechSense editor separates the meshes for the conductive
751 and non-conductive parts and generates separate .stl files for each of these components. The .stl files can be loaded into a
752 slicing software for 3D printing where each file gets assigned a non-conductive and a conductive filament, respectively.
753
754

755 6.2 3D Printing

756 To manufacture the mechanism with integrated sensors in one pass, users load both non-conductive and conductive
757 filament into a multi-material FDM 3D printer. Below, we provide more details on the conductive filament we used, the
758 3D printer hardware and print settings, as well as considerations regarding build plate adhesion.
759

760 **Conductive Material:** We use Electrifi filament from Multi3d¹ since it has the highest conductivity (0.006 ohm.cm)
761 among commercially available conductive thermoplastic filaments to date.
762

763 **3D Printer Hardware and Print Settings:** We use an Ultimaker S5 3D printer with a 0.6mm CC printcore from
764 Ultimaker to accommodate the Electrifi conductive filament. The Electrifi filament is considerably softer than regular
765 PLA filament and thus produces better print qualities with the use of a larger, abrasive material resistant printcore. To
766 avoid grinding the softer conductive material during extrusion, we manually set the distance between the filament
767 gears to the lowest feeder tension for our printer.
768
769

770 **Build Plate Adhesion:** Electrifi filament does not adhere well to glass substrates such as the print platform of our
771 3D printer. Thus, we first print a layer of PLA on which the conductive traces can be reliably printed with sufficient
772 adhesion. This first layer is automatically generated by our MechSense plugin for SolidWorks. (Figure 12).
773

774 **Nozzle Print Speed:** We noticed that the Electrifi filament can smudge easily during printing. In particular, if two
775 conductive areas, like the sensor patches, are close to each other, smudging can lead to short circuits between adjacent
776 patches. This material behavior can be attributed to the low melting temperature of Electrifi, which can render it to be
777

778 ¹<https://www.multi3d.com/product/electrifi/>
779
780

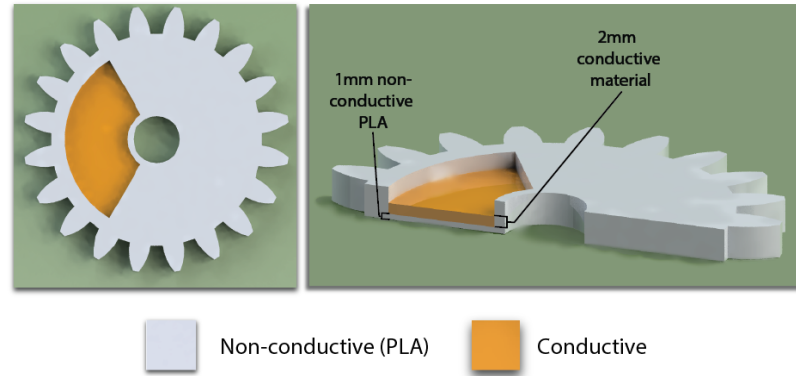


Fig. 12. Build plate adhesion can be improved by printing a layer of PLA filament underneath the conductive filament.

very soft if it is not given enough time to cool down sufficiently. To alleviate print failures due to smudging, we reduced our print speeds from 15 mm/s to 7mm/s when printing the sensor patches, wires, and the floating capacitor to allow the conductive filament to cool down before a new layer is printed on top.

6.3 Connecting Sensors and Streaming Data

Users connect their 3D printed mechanisms to the sensing board and upload code to the microcontroller that streams the raw sensor values to the serial port.

Connecting Sensors to the Sensing Board: Since Electrifi has a high contact resistance, it is difficult to connect wires to printed traces just by taping them on. Instead, we integrate wires directly into the conductive material of the sensors by heating up the tip of the wires with a soldering iron and pushing them into the Electrifi traces. The hot tip melts the filament which allows us to push the wires in. This technique brings a large part of the wire's surface in contact with the conductive filament. After cooling down, the wires are tightly and reliably connected to the printed sensor.

Uploading Code and Retrieving Sensor Values: Users next upload our code to a microcontroller that is connected to a capacitive sensing board (FDC2214). Our code collects the read sensor values from the board using an open source library² to read data from the sensing board and streams them with a time stamp to a computer and our Processing UI via the serial port. The FDC2214 board samples at a rate of 27ms for all sensor channels. **Sensing Board:** We use a resonance-based capacitive sensor board that utilizes an RLC circuit (FDC2214, \$50). This board has four sequential channels for capacitive sensing, with a capacitive sensing resolution of up to 28 bits (range: $1pF$ to $250nF$), and uses 3.3V logic.

6.4 MechSense UI to Process Raw Sensor Data

To make MechSense accessible to a wide range of users, we developed a Java/Processing implementation that supports users in performing the calibration step and converts the raw sensor data into angular position, direction of rotation, and rotational speed. To do the initial calibration, the user clicks on the button "Start Calibration". This loads a progress

²<https://www.arduino.cc/reference/en/libraries/fdc2214>

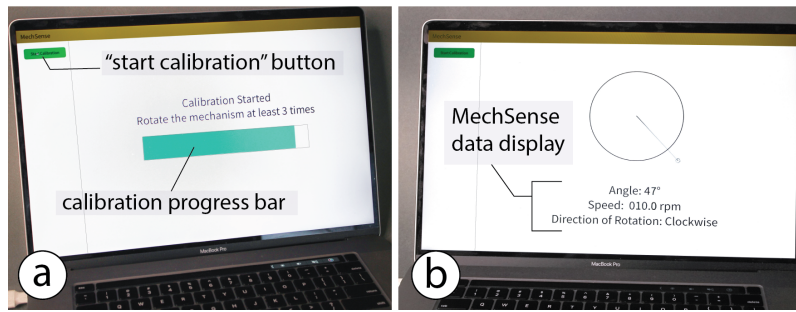


Fig. 13. Calibrating and viewing MechSense sensor data through MechSense Sensor UI

bar that guides users to do 3 rotations. The system can track the progress by counting the global minima. When the calibration is completed, it reads the streamed data from the sensors and displays the angle, speed, and direction of rotation on the screen. The processing of the raw sensor data is implemented as a Java library that can be imported to any Java program and enables users to leverage MechSense for customized applications.

7 TECHNICAL EVALUATION

We ran a technical evaluation to determine the error in the angular position estimation for different spacing between sensor patches, the effect of different sensor patch sizes, and the influence of capacitive objects (such as the user's hand) on the sensing accuracy. We compared all angular position estimation to a commercial rotational magnetic encoder (14 bit encoder, AS5048) that acts as our ground truth.

7.1 Spacing Between Sensors

To determine the influence of the separation distance between the sensor patches on the angular sensing accuracy, we conducted an experiment that evaluated the angular position estimations with sensor patches that have a separation of 3mm, 5mm, and 7mm.

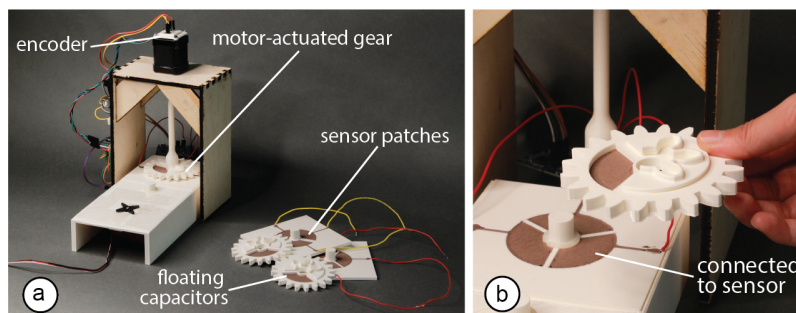


Fig. 14. Evaluation Experiment Setup. (a) We connected a stepper motor with an integrated rotational encoder to a MechSense gear with an integrated floating capacitor. The gear was mounted on a base plate with three sensor patches. (b) The configuration of the sensor patches and the rotating element varied depending on the experiment configuration. We used the rotational encoder (14 bit encoder, AS5048) in the stepper motor (NEMA 17, 1.8° step size) as the ground truth.

Apparatus: We printed a contraption that can hold a removable base plate with three sensor patches. We also printed a shaft in the middle of the sensor patches that can hold a removable rotating gear with an integrated floating capacitor. The gear was held at a constant distance from the sensor patches by adding a 1mm thick washer between the gear and the base plate. We connected the gear to an axle that was mounted to a stepper motor (NEMA 17, 1.8° step size) (Figure 14a). We evaluated 3 different spacings between sensor patches by printing 3 base plates with sensor patches separated by 3mm, 5mm, and 7mm (Figure 15) while keeping the total surface area constant at 765mm^2 . We also printed 3 matching gears whose floating capacitor area corresponded to the sensor patch area on the base plate. We printed each plate-gear pair three times to capture variations in the sensing accuracy caused by quality of the 3D print (3 conditions x 3 prints = 9 plate-gear pairs in total). For each plate-gear pair, we first connected all three sensor patches to our sensing board and then placed each of the base plates and the matching rotating gears into the contraption.

We also attempted to print a plate with 1mm separation but noticed that, due to the printing resolution of our current 3D printer, the conductive material layers were not perfectly separated and created a short circuit between plates.

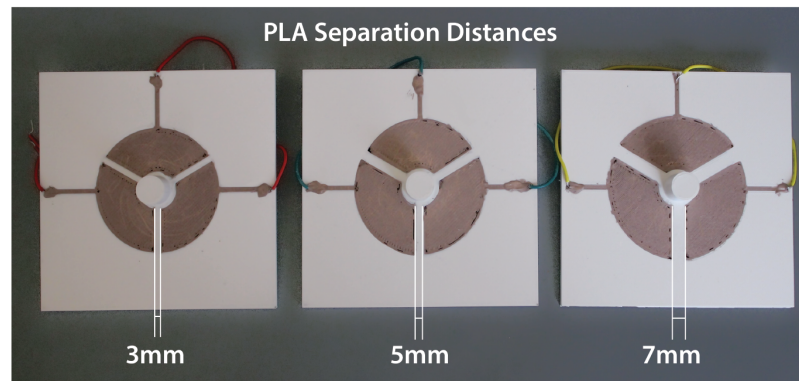


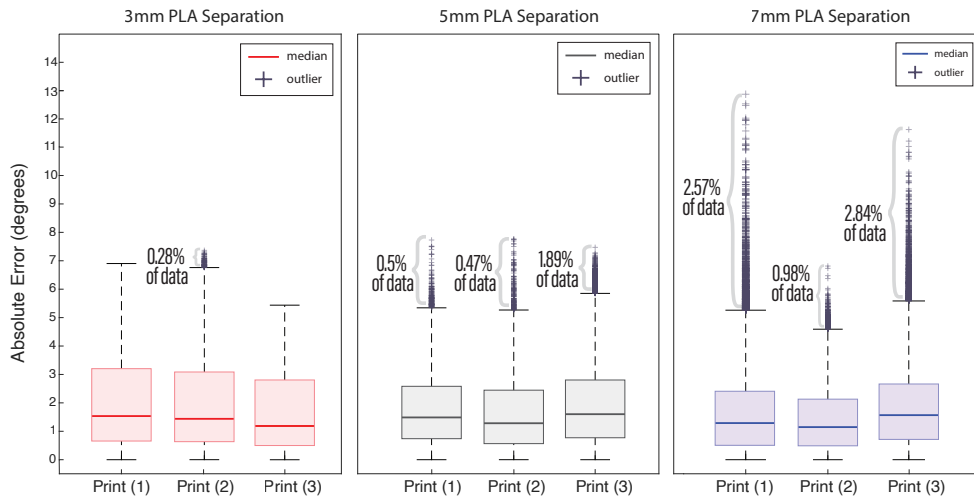
Fig. 15. Separation distances between sensor patches ast 3mm, 5mm, and 7mm.

Procedure: We first calibrated each sensor by rotating the gear three times to generate the extrema estimations. After that, we captured the raw sensor data for each plate-gear pair for 50 rotations at a speed of 10 RPM with micro-stepping, and used our computational model to convert the read data into the angular position estimation. Finally, we compared our estimate to the ground truth of the rotational encoder in the connected stepper motor and computed the error.

Results: Figure 16 shows the results of the experiment. All 9 plate-gear pairs show similar median errors between 1.1°-1.5°. The 75th Percentile of all captured error values reached up to 3% error across all experiment conditions, and the box plot whiskers (which are at the value of the 75th Percentile + 1.5 * range between the 25th and 75th Percentile) reached up to 7° for the 3mm PLA separation, up to 5.9° for the 5mm PLA separation, and up to 5.5° for the 7mm PLA separation. However, we found an increasing amount of outliers for larger PLA separations. While the 3mm PLA separation had nearly no outliers, with one print having 0.28% of the captured data in the outlier range and an error of up to 7°, the 5mm PLA separation prints had a maximum of 1.89% of sensor values as outliers within a single print, and a maximum error of approximately 8°. The 7mm print's error value reached a maximum of around 13° for 2.57% of its sensor data as outliers. This increasing error can be explained by the change in the signal's shape for larger PLA separations.

937 Figure 17 shows the sensor signal for 3mm, 5mm, and 7mm PLA separation overlaid on top of each other. As the
 938 distance between adjacent sensor patches is increased, the shape of the signal becomes wider at the maxima (where the
 939 floating capacitor has a 50% overlap with two sensor patches). Thus, our approximation polynomial, which we derived
 940 from a sensor that had 3mm separation, does not approximate the signal of the 5mm and 7mm separation samples
 941 well near the maxima. Since the change in signal shape is still small, it only mildly effects the mean error value of our
 942 angular position estimation and leads to a detectable increase in outliers.
 943
 944

945
 946
 947



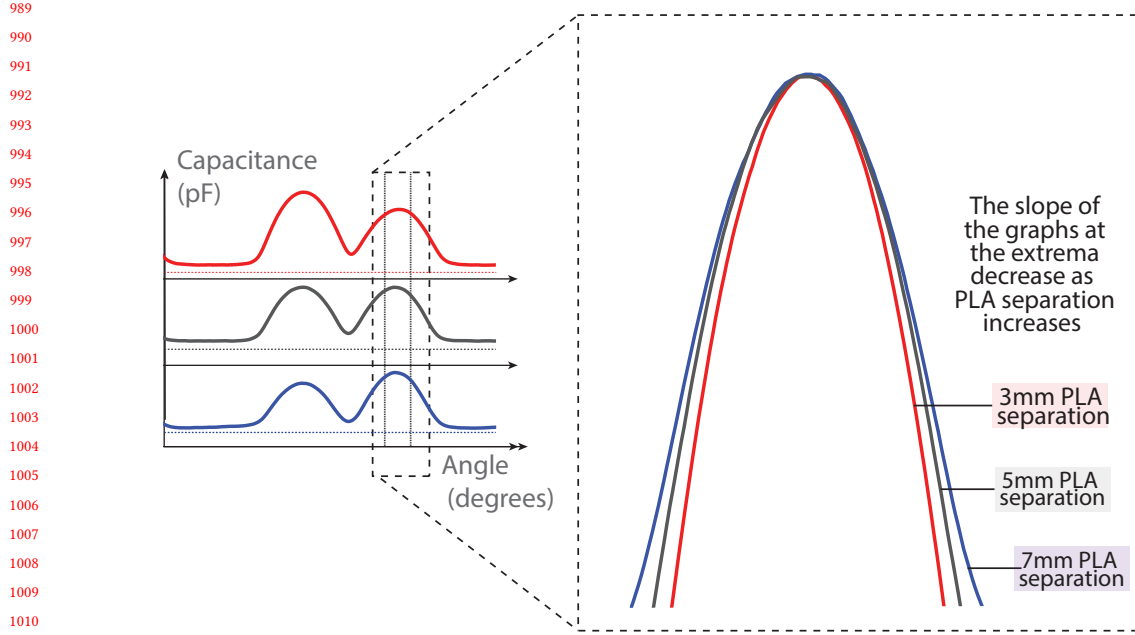
967 Fig. 16. Error values retrieved from the angular position estimation for 50 rotations across 3 different prints for 3mm, 5mm, and 7mm
 968 PLA separation gaps between adjacent patches. The data demonstrates similar median error values (1.1°- 1.5°), but also shows an
 969 increase in the number of outlier values as the separation distance increases.
 970

971
 972

973 7.2 Effect of Sensor Size on Sensing Accuracy

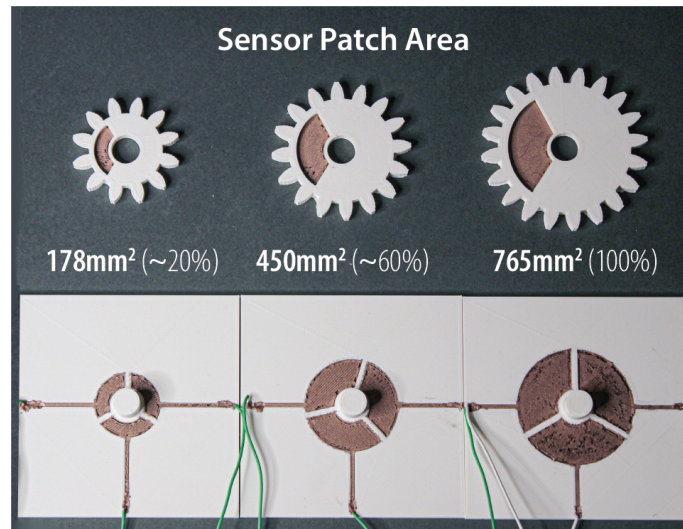
974 To determine the influence of sensor patch size on the angular position estimation error, we conducted an experiment that
 975 evaluated the angular position estimation with sensor patches that had a surface area of $765mm^2$ (100%), $450mm^2$ (60%),
 976 and $178mm^2$ (20%).
 977

978
 979 **Apparatus:** We used the same experimental setup as in section 7.1, but this time we printed gears of different sizes
 980 with corresponding smaller sensor patch surface areas. We evaluated 3 different sizes: $765mm^2$ (100%), $450mm^2$ (60%),
 981 and $178mm^2$ (20%) (Figure 18). We used a 3mm separation distance between sensor patches. We printed gears and base
 982 plates with matching floating capacitor and sensor patch areas three times for each combination (3 conditions x 3 prints
 983 = 9 plate-gear pairs in total). Since the $765mm^2$ sized gear with 3mm separation between sensor patches is identical to
 984 the experiment setup of section 7.1 for the 3mm separation case, we reused the captured sensor data in this experiment
 985 and compare it to the sensor patches with $450mm^2$, and $178mm^2$ patch surface area.
 986
 987
 988



1013 Fig. 17. Comparison of the signal shape at a maximum for 3mm, 5mm, and 7mm PLA separation. We see a slight decreasing slope of
1014 the signal for increasing PLA separation distances.

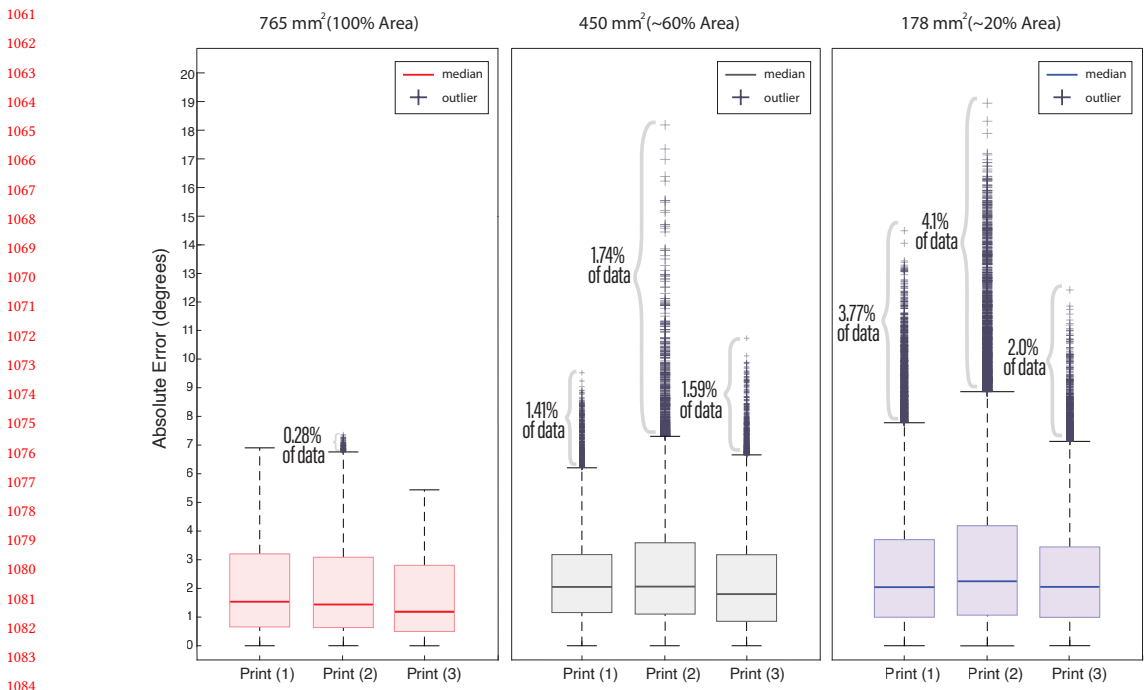
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040



1035 Fig. 18. Different sensor patches sizes. We evaluated the angular position estimation error for sensors with a total surface area of
1036 178mm^2 (20%), 450mm^2 (60%), and 765mm^2 (100%).

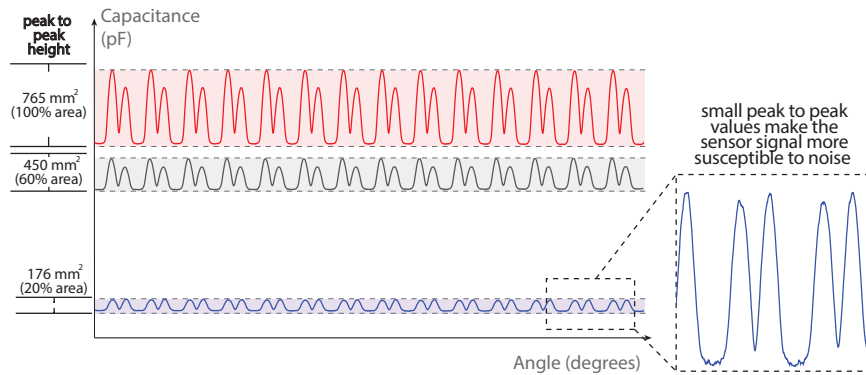
1041 **Procedure:** For each plate-gear pair we conducted an initial calibration by rotating the gears three times to get an
 1042 estimation for the extrema. Finally, we rotated the gears at 10 RPM for 50 rotations each. We recorded the signal for
 1043 each of the sensor patches and used our computational model to convert the raw sensor data into the angular position
 1044 estimation. Finally, we used the rotational encoder in our stepper motor to generate the ground truth that we compare
 1045 the sensor values to.
 1046
 1047

1048 **Results:** Figure 19 shows the results of the experiment. We found that the mean error across all conditions was similar
 1049 with $1.4^\circ(765mm^2)$, $2.0^\circ(450mm^2)$, and $2.1^\circ(178mm^2)$. However, we also observed a growing amount of outliers for smaller
 1050 sensor sizes where the $450mm^2$ showed in the worst case 1.74% of the sensor data being outliers with a maximum
 1051 error of 18.2° , and for the $178mm^2$ we found 4.1% of the sensor data being outliers with up to 18.9° error in the worst
 1052 case. The increasing error for smaller sensor sizes can be explained by the change of signal strength of the captured
 1053 data. Figure 20 shows the sensor data for multiple rotation for all three sensor sizes. One can see that the amplitude of
 1054 the signal decreases for smaller sensor sizes. This makes the sensor values at extrema less distinct, i.e. the change in
 1055 capacitance becomes smaller, and may lead to misdetections at the extrema.
 1056
 1057
 1058
 1059
 1060



1086
 1087 Fig. 19. Angular position estimation error for sensor patches of $765mm^2$, $450mm^2$, $178mm^2$ surface area. The median error stays in
 1088 a similar range to the $765mm^2$ sensor patch with a slight increase to 1.9° ($450mm^2$), and 2.1° ($178mm^2$). However, the amount of
 1089 outliers with errors up to 18° increase for 1.74% of the sensor values for the $450mm^2$ patch, and up to 4.1% of the sensors, with errors
 1090 reaching 19° , for the $178mm^2$ patch in the worst case.
 1091
 1092

1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108



1109
1110
1111
1112
1113

Fig. 20. Sensor Signal for patches with a total area of 765mm^2 , 450mm^2 , and 176mm^2 . Smaller sensor patches produce a smaller overall capacitance which makes them susceptible to external noise. For example, the extrema of the sensor patch with 176mm^2 show a more fluctuations than the extrema of the 765mm^2 patches.

1114

7.3 Effect of User's Hand Proximity on MechSense

1115

We evaluated the angular position estimation error with a user's hand at 5 different distances from the MechSense encoder.

1116

1117

1118

1119

1120

1121

1122

1123

1124

1125

1126

1127

1128

1129

1130

1131

1132

1133

1134

1135

1136

1137

1138

1139

1140

1141

1142

1143

1144

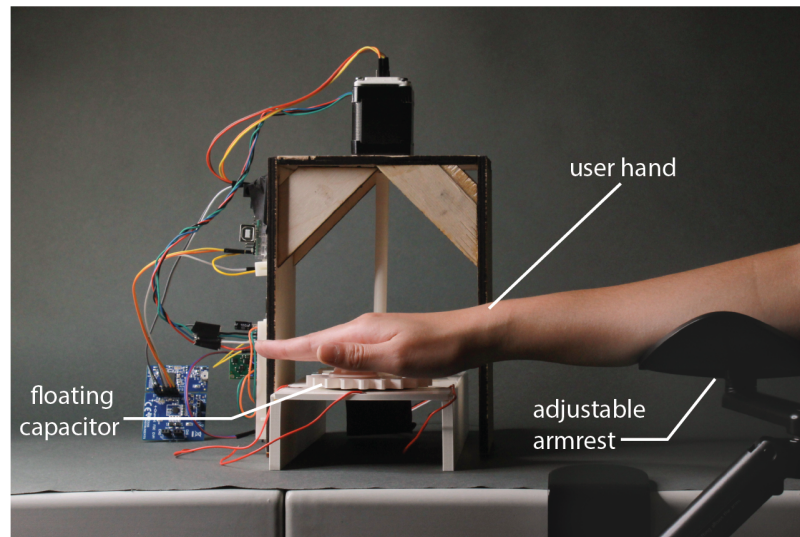


Fig. 21. Experiment setup: user hand placed at different heights on top of rotating gear apparatus.

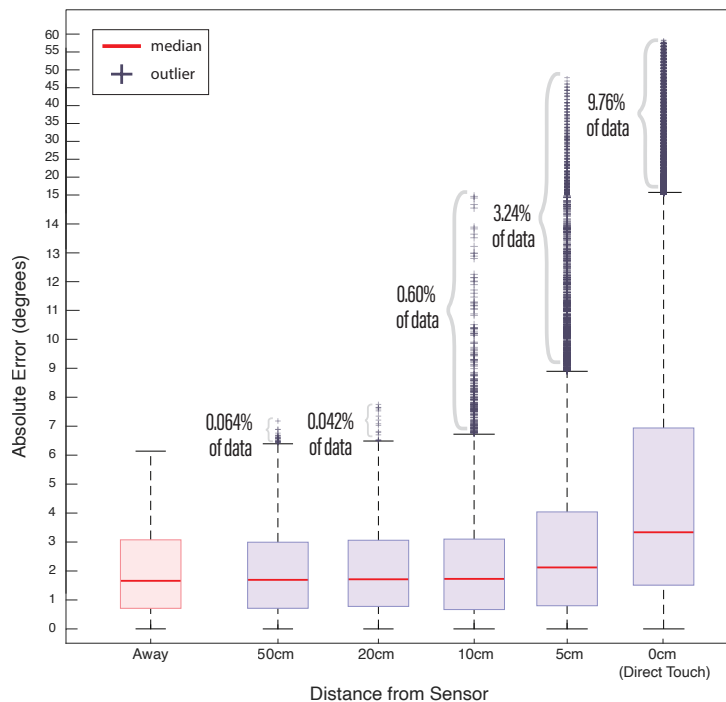
Apparatus: We used the same setup as in the previous evaluations but mounted a height-adjustable arm rest next to the MechSense encoder (Figure 21). We added the arm rest to provide the user a stable support to keep their hand at a

Manuscript submitted to ACM

1145 constant distance from the MechSense encoder over a longer period of time. The sensor patches had an area of $765mm^2$
 1146 and a PLA separation of 3mm between each patch.
 1147

1148 **Procedure:** We calibrated the sensor by rotating the gear three times without a user’s hand in proximity. Next, we
 1149 captured the angular position estimation error by rotating the gear 50 times at 10RPM with a user’s hand at a fixed
 1150 distance from the MechSense encoder. We asked a participant to place their hand on the arm rest which we set to a
 1151 custom height and let them hover with their hand above the rotating gear. We repeated this procedure for 5 gear-hand
 1152 distances: 50cm, 20cm, 10cm, 5cm, and 0cm (direct touch on the base plate). We measured the distance between the
 1153 hand and the gear with a ruler and asked the participant to keep the hand as steady as possible. Each experiment took
 1154 5 minutes. In addition, we ran one experiment with no hand in proximity to the MechSense encoder to generate a
 1155 baseline that allowed us to compare the error values with a hand in proximity.
 1156
 1157
 1158

1159 **Results:** We observe a similar mean error when the user’s hand is at a distance of 50cm, 20cm, and 10cm from the
 1160 MechSense encoder.
 1161
 1162
 1163
 1164



1190 Fig. 22. Angular position estimation error with a user’s hand in 5 distances above the rotating gear: 50cm, 20cm, 10cm, 5cm, and
 1191 directly touching the 3D printed base, and one additional experiment with no user in proximity as a baseline. We observed minimal
 1192 impact of a user’s hand between 50cm to 10cm distance similar to our baseline. However, we noticed an increasing mean error and a
 1193 larger amount of outlier for 5cm distance and for direct touch.
 1194
 1195
 1196

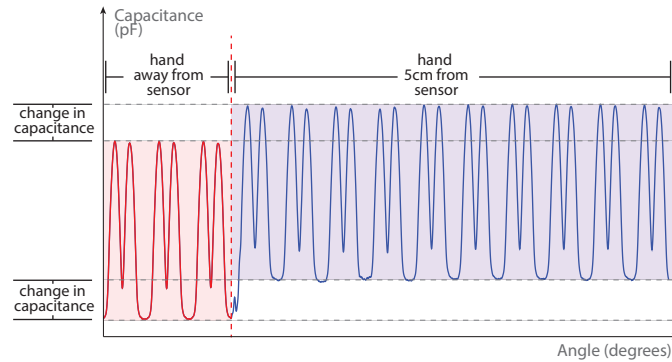


Fig. 23. Capacitance change in presence of a user’s hand. We observe a global increase in capacitance in presence of the user’s hand (5cm distance). The capacitance change leads to an increase in angular position estimation error in particular in the first few rotations until our system updated the new values for the extrema and our polynomial fitting.

The amount of outliers in these cases also remains small with 0.064% (50cm), and 0.042% (20cm) of the sensor values being outlier with a maximum error of almost 8° . The 10cm distance had a slightly larger amount of outliers (0.6%) with a maximum error of 15.6° . However, the mean error and amount of outliers increased when the user’s hand was at a distance of 5cm and directly touching the base with the sensor patches. We measured a mean error of 2.1 (5cm) and 3.3 (direct touch). The amount of outliers increased to 3.24% (maximum error: 47.8) of the sensor data for 5cm, and to 9.76% (maximum error: 58.3) of the total sensor data when directly touching the base with the sensor patches. This indicates that MechSense encoders experience increased angular position estimation errors in the presence of a user’s hand below 10cm distance. This is, however, expected as a user’s hand introduces noise by changing the capacitance of the sensor system. Figure 23 shows the change in capacitance when a user’s hand is present at 5cm distance. The capacitance values show a global increase that leads to detection errors of the extrema in our computational model. However, the error rate decreases as our model updates the magnitude of the extrema after several rotations to the new signal values. Thus, the majority of outliers reported in Figure 22 originate from the first few rotations until our system can compensate for the capacitance change.

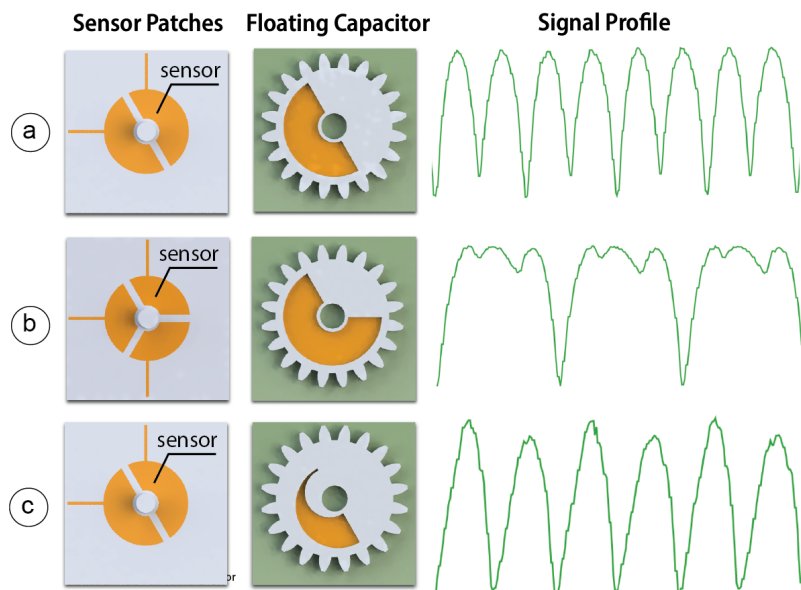
8 LIMITATIONS AND FUTURE WORK

We next discuss limitations of our work and potential avenues for future research.

Generating Conductive Traces: In our current user interface, the user has to manually route the traces, i.e. draw the path from the sensor patch to a location where they would like to connect the trace of the sensor patch to the sensing board. For future work, we plan to auto-route the traces.

Other Sensor Layouts: We also experimented with alternative sensor layouts (Figure 24). The first alternative sensor layout used only two sensor patches (Figure 24a) and created a signal profile similar to the three sensor patch layout. It only required two sensors on the sensing board, which allowed us to connect more mechanisms without requiring

1249 additional multiplexing. One drawback is that this sensor layout does not allow us to sense the direction of rotation
 1250 since three sensor signals are required to eliminate ambiguity. The second alternative sensor layout (Figure 24b) used
 1251 a double-sized floating capacitor on the moving part of the mechanism. This produces a signal with three maxima
 1252 corresponding to 25%, 50%, and 25% overlap as the floating capacitor moves across the sensor patch. For future work,
 1253 we plan to further explore this sensor patch design to extract additional features based on slopes and intersections.
 1254 Finally, we considered a gradient sensor layout (Figure 24c) in which the floating capacitor area increases. This allows
 1255 us to extract the direction of rotation from one sensor patch only. However, we found that 3D printing the thin part of
 1256 the gradient sensor patch was difficult to achieve when the rotating geometry was small.
 1257
 1258
 1259
 1260
 1261
 1262
 1263
 1264



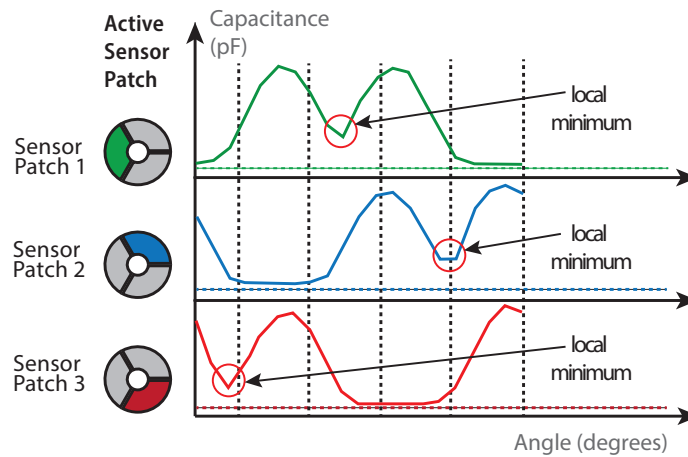
1265
 1266
 1267
 1268
 1269
 1270
 1271
 1272
 1273
 1274
 1275
 1276
 1277
 1278
 1279
 1280
 1281
 1282
 1283
 1284
 1285
 1286 Fig. 24. Alternative sensor layouts: (a) two sensor patches on the static part, (b) double-sized floating capacitor on the moving part,
 1287 (c) gradient floating capacitor on the moving part.
 1288
 1289

1290 **Simulation of Sensor Signals:** While we only used the sensor signal simulation in section 4.2 to confirm the waveform
 1291 of our chosen sensor layout, we can also use the simulation to predict the sensor signal of arbitrary geometries and
 1292 arrangements of sensor patches. It thereby allows for rapid exploration of the parameter space, including the sensor
 1293 patch shape and size and the dielectric strengths of the insulating material, which we could use in future work to
 1294 enhance our computational prediction model for angular position accuracy.
 1295
 1296

1297 **Compensating for Proximity of a User's Hand:** We showed in our technical evaluation that the presence of a user's
 1298 hand can introduce high errors in the first few rotations before our system can compensate for the change in capacitance
 1299 induced by the user's hand. In future work, we want to compensate for such noise instantly by taking into account the
 1300

1301 change in capacitance for our global minimum. Since the increase in capacitance in proximity to a user’s hand behaves
 1302 similar to a global lift up of the original sensor signal, it might be possible to estimate this lift up by computing the
 1303 difference of the global minimum value that we got from the initial calibration and the actual global minimum when
 1304 the user’s hand is present. Finally, we subtract this difference from all sensor values to obtain a signal that is close to
 1305 our original signal. Since at least one sensor patch is in the global minimum at any time, this compensation could be
 1306 computed almost instantly.
 1307
 1308

1309 **Sampling Rate & Speed Limitation:** Our sensing board has a sampling rate of 27ms per sample across all sensor
 1310 channels which generates sensor data that is fairly close to the true capacitance of the system at low rotation speed.
 1311 However, as the speed of rotation increases, the produced sensor data profile becomes less smooth with multiple linear
 1312 segments. Figure 25 shows a sensor profile at 200RPM. One can see that not all extrema are perfectly captured at that
 1313 rotation speed. These inaccuracies will introduce errors in our polynomial fitting and lead to an increasing angular
 1314 position estimation error at a higher rotation speed. Such application scenarios that require high rotation speed will
 1315 also require to utilize a sensing board that offers a higher sampling rate.
 1316
 1317
 1318
 1319
 1320



1339 Fig. 25. Sampling of the sensor signal at 200RPM at a sampling rate of 27ms. As less samples become available for a full rotation,
 1340 the sensor signal becomes less smooth showing rather linear segments. This deviation from our approximation polynomial will
 1341 lead to a higher angular position estimation error. Application scenarios that require a high rotation speed will also require using a
 1342 sensing boards that supports a higher sampling rate.
 1343
 1344

1345 9 CONCLUSION

1347 In this paper, we investigated how to integrate sensing into rotating mechanisms via conductive multi-material 3D
 1348 printing to enable them to sense their direction of rotation, speed, and angular position. We showed how a sensor
 1349 layout that integrates sensors with the static part of a mechanism and a floating capacitor with the moving part of the
 1350 mechanism generalizes across different rotational mechanisms, such as gears, linkages, and wheels. We presented an
 1351

editor that facilitates the integration of the sensors with the mechanism geometry, and that exports the 3D printable files. We also contribute a Java/Processing tool that uses our computational model to convert the raw sensor data into angular position estimation, direction and speed of rotation. We evaluated the angular position estimation error for different spacing between sensors, the size of the sensor patches, and the influence of the proximity of a user's hand near a MechSense encoder. For future work, we plan to explore how to increase the robustness of our sensing method to external noise, and develop 3D printable sensors that can monitor other types of mechanisms.

REFERENCES

- [1] Jesse Burstyn, Nicholas Fellion, Paul Strohmeier, and Roel Vertegaal. 2015. PrintPut: Resistive and Capacitive Input Widgets for Interactive 3D Prints. In *Human-Computer Interaction – INTERACT 2015*, Julio Abascal, Simone Barbosa, Mirko Fetter, Tom Gross, Philippe Palanque, and Marco Winckler (Eds.). Springer International Publishing, Cham, 332–339.
- [2] V. Ferrari, A. Ghisla, D. Marioli, and A. Taroni. 2004. Capacitive angular-position sensor with electrically-floating conductive rotor and measurement redundancy. In *Proceedings of the 21st IEEE Instrumentation and Measurement Technology Conference (IEEE Cat. No.04CH37510)*, Vol. 1. 195–200 Vol.1. <https://doi.org/10.1109/IMTC.2004.1351027>
- [3] P.L. Fulmek, F. Wandling, W. Zdiarsky, G. Bresseur, and S.P. Cermak. 2002. Capacitive sensor for relative angle measurement. *IEEE Transactions on Instrumentation and Measurement* 51, 6 (2002), 1145–1149. <https://doi.org/10.1109/TIM.2002.808052>
- [4] M. Gasulla, Xijun Li, G.C.M. Meijer, L. van der Ham, and J.W. Spronck. 2002. A contactless capacitive angular-position sensor. In *SENSORS, 2002 IEEE*, Vol. 2. 880–884 vol.2. <https://doi.org/10.1109/ICSENS.2002.1037224>
- [5] Ben Greenspan, Eric M. Gallo, and Andreea Danielescu. 2022. FlexKeys: Rapidly Customizable 3D Printed Tactile Input Devices with No Assembly Required. <https://doi.org/10.48550/ARXIV.2203.00757>
- [6] Liang He, Jarrid A. Wittkopf, Ji Won Jun, Kris Erickson, and Rafael Tico Ballagas. 2022. ModElec: A Design Tool for Prototyping Physical Computing Devices Using Conductive 3D Printing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 4, Article 159 (dec 2022), 20 pages. <https://doi.org/10.1145/3495000>
- [7] Bo Hou, Zhang Tian, Cao Li, Qi Wei, Bin Zhou, and Rong Zhang. 2017. A capacitive rotary encoder with a novel sensitive electrode. In *2017 IEEE SENSORS*. 1–3. <https://doi.org/10.1109/ICSENS.2017.8234143>
- [8] Vikram Iyer, Justin Chan, and Shyamath Gollakota. 2017. 3D Printing Wireless Connected Objects. *ACM Trans. Graph.* 36, 6, Article 242 (nov 2017), 13 pages. <https://doi.org/10.1145/3130800.3130822>
- [9] Cedric Honnet Jack Forman Stefanie Mueller. Jun Gong, Olivia Seow. 2021. *MetaSense: Integrating Sensing Capabilities into Mechanical Metamaterial*. Association for Computing Machinery, New York, NY, USA.
- [10] M. Karali, A. T. Karasahin, O. Keles, M. Kocak, and M. A. Erisim. 2018. A new capacitive rotary encoder based on analog synchronous demodulation. *Electrical Engineering* 100 (2018), 975–983. <https://doi.org/10.1007/s00202-018-0677-9>
- [11] Kunihiko Kato, Kaori Ikematsu, and Yoshihiro Kawahara. 2020. CAPATH: 3D-Printed Interfaces with Conductive Points in Grid Layout to Extend Capacitive Touch Inputs. *Proc. ACM Hum.-Comput. Interact.* 4, ISS, Article 193 (Nov. 2020), 17 pages. <https://doi.org/10.1145/3427321>
- [12] Danny Leen, Raf Ramakers, and Kris Luyten. 2017. StrutModeling: A Low-Fidelity Construction Kit to Iteratively Model, Test, and Adapt 3D Objects. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (Québec City, QC, Canada) (UIST '17)*. Association for Computing Machinery, New York, NY, USA, 471–479. <https://doi.org/10.1145/3126594.3126643>
- [13] Karola Marky, Martin Schmitz, Verena Zimmermann, Martin Herbers, Kai Kunze, and Max Mühlhäuser. 2020. 3D-Auth: Two-Factor Authentication with Personalized 3D-Printed Items. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (Honolulu, HI, USA) (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3313831.3376189>
- [14] Karola Marky, Andreas Weiß, Florian Müller, Martin Schmitz, Max Mühlhäuser, and Thomas Kosch. 2021. Let's Frets! Mastering Guitar Playing with Capacitive Sensing and Visual Guidance. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems (Yokohama, Japan) (CHI EA '21)*. Association for Computing Machinery, New York, NY, USA, Article 169, 4 pages. <https://doi.org/10.1145/3411763.3451536>
- [15] Thijs Jan Roumen, Willi Müller, and Patrick Baudisch. 2018. Grafter: Remixing 3D-Printed Machines. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (Montreal QC, Canada) (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3173574.3173637>
- [16] Valkyrie Savage, Andrew Head, Björn Hartmann, Dan B. Goldman, Gautham Mysore, and Wilmot Li. 2015. Lamello: Passive Acoustic Sensing for Tangible Input Components. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (Seoul, Republic of Korea) (CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1277–1280. <https://doi.org/10.1145/2702123.2702207>
- [17] Martin Schmitz, Martin Herbers, Niloofar Dezfouli, Sebastian Günther, and Max Mühlhäuser. 2018. Off-Line Sensing: Memorizing Interactions in Passive 3D-Printed Objects. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (Montreal QC, Canada) (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–8. <https://doi.org/10.1145/3173574.3173756>
- [18] Martin Schmitz, Mohammadreza Khalilbeigi, Matthias Balwierz, Roman Lissermann, Max Mühlhäuser, and Jürgen Steimle. 2015. Capricate: A Fabrication Pipeline to Design and 3D Print Capacitive Touch Sensors for Interactive Objects. In *Proceedings of the 28th Annual ACM Symposium*

- 1405 *on User Interface Software and Technology* (Charlotte, NC, USA) (*UIST '15*). Association for Computing Machinery, New York, NY, USA, 253–258.
1406 <https://doi.org/10.1145/2807442.2807503>
- 1407 [19] Martin Schmitz, Florian Müller, Max Mühlhäuser, Jan Riemann, and Huy Viet Viet Le. 2021. Itsy-Bits: Fabrication and Recognition of 3D-Printed
1408 Tangibles with Small Footprints on Capacitive Touchscreens. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*
1409 (Yokohama, Japan) (*CHI '21*). Association for Computing Machinery, New York, NY, USA, Article 419, 12 pages. [https://doi.org/10.1145/3411764.](https://doi.org/10.1145/3411764.3445502)
1410 3445502
- 1411 [20] Martin Schmitz, Jürgen Steimle, Jochen Huber, Niloofar Dezfuli, and Max Mühlhäuser. 2017. Flexibles: Deformation-Aware 3D-Printed Tangibles for
1412 Capacitive Touchscreens. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) (*CHI '17*).
1413 Association for Computing Machinery, New York, NY, USA, 1001–1014. <https://doi.org/10.1145/3025453.3025663>
- 1414 [21] Tatyana Vasilevitsky and Amit Zoran. 2016. Steel-Sense: Integrating Machine Elements with Sensors by Additive Manufacturing. In *Proceedings of*
1415 *the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) (*CHI '16*). Association for Computing Machinery, New
1416 York, NY, USA, 5731–5742. <https://doi.org/10.1145/2858036.2858309>
- 1417 [22] Hewen Wang, Kai Peng, Xiaokang Liu, Zhicheng Yu, and Ziran Chen. 2021. Design and Realization of a Compact High-Precision Capacitive Absolute
1418 Angular Position Sensor Based on Time Grating. *IEEE Transactions on Industrial Electronics* 68, 4 (2021), 3548–3557. [https://doi.org/10.1109/TIE.2020.](https://doi.org/10.1109/TIE.2020.2977540)
1419 2977540
- 1420 [23] Clement Zheng, Jee Eun Kim, Daniel Leithinger, Mark D. Gross, and Ellen Yi-Luen Do. 2019. Mechamagnets: Designing and Fabricating Haptic and
1421 Functional Physical Inputs with Embedded Magnets. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied*
1422 *Interaction* (Tempe, Arizona, USA) (*TEI '19*). Association for Computing Machinery, New York, NY, USA, 325–334. [https://doi.org/10.1145/3294109.](https://doi.org/10.1145/3294109.3295622)
1423 3295622
- 1424 [24] Dezhi Zheng, Shaobo Zhang, Shuai Wang, Chun Hu, and Xiaomeng Zhao. 2015. A Capacitive Rotary Encoder Based on Quadrature Modulation and
1425 Demodulation. *Instrumentation and Measurement, IEEE Transactions on* 64 (01 2015), 143–153. <https://doi.org/10.1109/TIM.2014.2328456>
- 1426
- 1427
- 1428
- 1429
- 1430
- 1431
- 1432
- 1433
- 1434
- 1435
- 1436
- 1437
- 1438
- 1439
- 1440
- 1441
- 1442
- 1443
- 1444
- 1445
- 1446
- 1447
- 1448
- 1449
- 1450
- 1451
- 1452
- 1453
- 1454
- 1455
- 1456