Itsy-Bits: Fabrication and Recognition of 3D-Printed Tangibles with Small Footprints on Capacitive Touchscreens

Martin Schmitz
schmitz@tk.tu-darmstadt.de
TU Darmstadt
Darmstadt, Germany

Florian Müller
mueller@tk.tu-darmstadt.de
TU Darmstadt
Darmstadt, Germany

Max Mühlhäuser
max@tk.tu-darmstadt.de
TU Darmstadt
Darmstadt, Germany

Jan Riemann
riemann@tk.tu-darmstadt.de
TU Darmstadt
Darmstadt, Germany

Huy Viet Le
mail@huyle.de
University of Stuttgart
Stuttgart, Germany

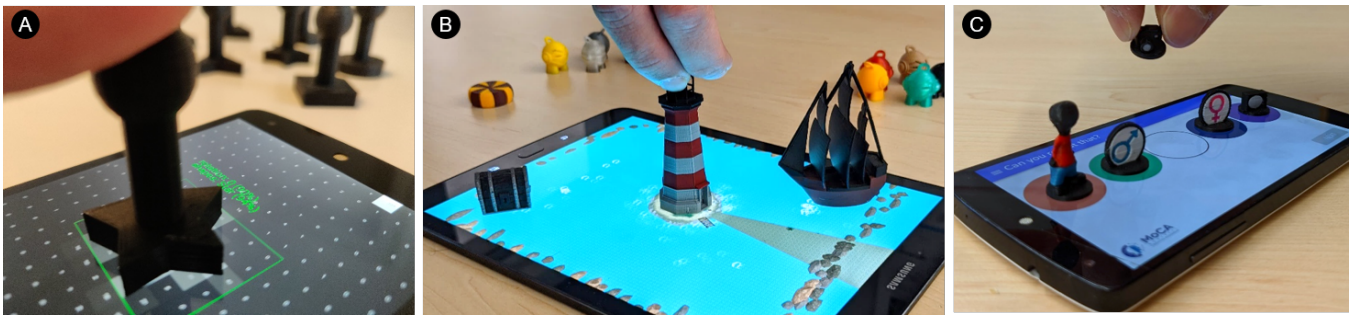

Figure 1: Itsy-Bits recognizes 3D-printed tangibles as small as a fingertip via the capacitive image of an embedded conductive shape (A). This opens up a variety of tangible user interfaces on the most common form factors of touchscreens, such as individualized interactive board games (B) or a more tangible learning experience (C).

ABSTRACT

Tangibles on capacitive touchscreens are a promising approach to overcome the limited expressiveness of touch input. While research has suggested many approaches to detect tangibles, the corresponding tangibles are either costly or have a considerable minimal size. This makes them bulky and unattractive for many applications. At the same time, they obscure valuable display space for interaction.

To address these shortcomings, we contribute Itsy-Bits: a fabrication pipeline for 3D printing and recognition of tangibles on capacitive touchscreens with a footprint as small as a fingertip. Each Itsy-Bit consists of an enclosing 3D object and a unique conductive 2D shape on its bottom. Using only raw data of commodity capacitive touchscreens, Itsy-Bits reliably identifies and locates a variety of shapes in different sizes and estimates their orientation. Through example applications and a technical evaluation, we demonstrate the feasibility and applicability of Itsy-Bits for tangibles with small footprints.

CHI '21, May 8–13, 2021, Yokohama, Japan

ACM ISBN 978-1-4503-8096-6/21/05…$15.00
https://doi.org/10.1145/3411764.3445502

CCS CONCEPTS

• **Human-centered computing** → **Interaction devices**.

KEYWORDS

Touchscreen; 3D Printing; Tangibles; Machine Learning

ACM Reference Format:
Martin Schmitz, Florian Müller, Max Mühlhäuser, Jan Riemann, and Huy Viet Le. 2021. Itsy-Bits: Fabrication and Recognition of 3D-Printed Tangibles with Small Footprints on Capacitive Touchscreens. In *CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3411764.3445502

1 INTRODUCTION

Today, touch-enabled devices are ubiquitous, ranging from smartphones and tablets to wall-sized displays. While the concept of touch is easy to understand, it is often criticized as lacking input expressiveness as it only encodes a single point of touch [53]. As one solution, research has proposed interactive tangible objects (in short, *tangibles*) that, when placed on a touchscreen, enable haptic control of on-screen contents by identifying the object together with its location and orientation [8, 26, 45, 59]. This strand of research has also proposed a great variety of interaction techniques that can be performed with tangibles, ranging from touch [51, 63] and deformation [3, 53, 60] to physical controls [19, 28, 62] and construction [5, 8, 33]. Remarkably, despite this large body of research,

the number of successful tangible interfaces (e.g., TangiPlay[1]) is relatively small compared to the multitude of touchscreens that have entered our everyday lives.

Although this trend is certainly influenced by many factors, one of the main reasons is their challenging trade-off between production costs and size: Tangibles are often either small but require costly hardware for active sensing or are comparably cheap to produce but bulky in size due to their use of spatially separated patterns of touchpoints required to operate on commodity capacitive touchscreens. The latter is particularly critical for the most widely used but smaller form factors (such as smartphones and tablets), as big tangibles hide valuable display space, which in turn reduces the benefits of a tangible user interface that aims to interweave physical and digital information.

To alleviate these limitations, we contribute Itsy-Bits: a fabrication pipeline and sensing approach for passive tangibles with a small footprint comparable to the size of a fingertip (16-20 mm [9]). Itsy-Bits are 3D printed in a single pass with varying embedded conductive shapes on their bottom. The type of their shape and their orientation are then sensed using only the low-resolution raw data provided by commodity capacitive touchscreens (see Figure 1A). To this end, we contribute a machine learning approach that distinguishes up to 30 different conductive shapes (10 shapes in three sizes between 12 × 12 mm and 20 × 20 mm) at accuracies of 95.63 % for shape and 98.57 % for size, as well as a mean rotation error of 6.53 °. This approach is based on a recorded data set that maps capacitive raw data of 3D-printed shapes to ground-truth postures.As illustrated by a set of example applications (see two of them in Figure 1B&C), Itsy-Bits enable new types of tangible geometries that do not require a large footprint and, hence, allow for a more direct tangible input as the user's view is less obstructed while aiming at a screen location.

In summary, the contributions of this paper are two-fold:

- A fabrication pipeline to create tangibles using an integrated design tool without further assembly.
- A sensing approach that reliably identifies and locates 10 shapes in three sizes (12 × 12 mm, 16 × 16 mm, 20 × 20 mm), including an estimate of their rotation.

2 RELATED WORK

This paper is situated in the areas of (1) extending the expressiveness of interaction on touchscreens, (2) digital fabrication of interactive objects, and (3) tangibles on touchscreens.

2.1 Extending the Expressiveness of Interaction on Touchscreens

Previous work has used capacitive raw data to extend the interaction with touchscreens for a wide range of use cases: Holz and Baudisch [16] recognize parts of the body (e.g., ears, fist) for authentication. Further previous work [12, 29, 31] has used them to recognize fingers and parts of the hand. Researchers have also extended the sensing space of capacitive sensors to the environment of devices [61] and their whole surface to predict touch positions on the front [40], detect swipe errors [41], and to enable the whole hand to perform input [30].

Prior work on machine learning with capacitive raw data either distinguish two sources of touch (e.g., palm vs. finger [29], left vs. right thumb [31]), or estimates the orientation of an index finger. In contrast, Itsy-Bits, to the best of our knowledge, is the first work that identifies tangibles based on a conductive shape on such low-resolution capacitive images. We go beyond finger and hand identification on capacitive images by enabling practitioners to quickly fabricate and sense tangible user interfaces, further enriching interaction on widespread capacitive touchscreens.

While there exist optical markers (such as ArUco [46]) or now commonplace QR codes, these approaches are not tailored to operate with capacitive images of only 6 PPI (i.e., only 3 × 3 to 5 × 5 4 mm-wide pixels for a single shape). These approaches require higher resolution and, even when printed conductive, would not work on a capacitive touchscreen since slight shifts and rotations of the marker would lead to a division of capacitance into adjacent cells, making the resulting raw data of such a marker too blurred for recognition.

2.2 Digital Fabrication of Interactive Objects

Embedding or attaching components to non-interactive objects is one approach to add interactive capabilities to 3D objects. This can be accomplished by attaching capacitive [47] or acoustic [43] sensors, or embedding cameras [48] or accelerometers [17]. Even though these approaches require only a few components, they imply additional effort or work only with hollow objects that can be opened after printing.

An emerging stream of research investigates how to embed customized interactive structures in 3D-printed objects. This includes adding interactive input and output functionalities in 3D-printed objects through light pipes [6, 64], by filling internal pipes with media post-print [49], or via pipes that transmit sound [28]. Other approaches print interactive objects using conductive spray [22] or plastic [7, 25, 32, 38, 50, 52, 54].

Building on this research, Itsy-Bits are created with an integrated design tool and then 3D printed in a single pass without assembly.

2.3 Tangibles on Touchscreens

Research has investigated localization and identification of tangibles on touchscreens via optical markers [5, 46, 63], magnetic sensor grids [27, 33–36], or localized NFC [57]. However, these approaches require either bulky setups or additional sensing hardware.

To alleviate these limitations, research investigates how common capacitive touchscreens can be used to detect tangibles based on early work by Rekimoto [45]. By embedding conductive material or adding electronics to the tangible, the capacitive touch sensor can detect presence and location of tangibles using spatial touchpoint patterns [20, 58, 59, 66] and embedded passive-resistive components [21], determine combinations of multiple objects [8], or forward the touch on the object to the touchscreen [8, 24, 25]. While innovative, tangibles still have to be lavishly created and assembled.

Therefore, a recent stream of research uses 3D printing to ease the individual creation of inherently specialized tangibles. Many approaches combine conductive and insulating printing materials

[1] tangiplay.com

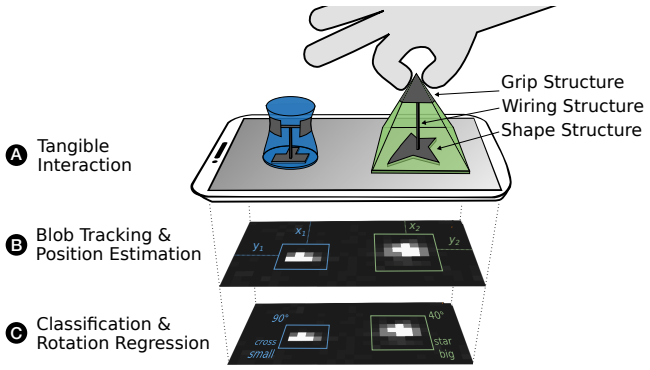

Figure 2: Based on the user's interaction (A), Itsy-Bits identifies and tracks blobs in the capacitive image (B) and classifies the conductive shape, its size, and its orientation using machine learning (C).

in such a way that the objects create a unique pattern of individual touchpoints to differentiate objects [13, 51, 53]. These approaches reduce the assembly effort typically required for embedding electrical structures in tangibles.

However, all touchpoints in such a pattern need to be sufficiently spaced (at least 11 mm according to [13]) to not get fused to a single touchpoint by the touch controller. As a sufficient number of differentiable tangibles requires many touchpoints, this either results in bulky tangibles or severely limits the number of distinguishable tangibles, when a small footprint is required. Comparable approaches, such as PERCs [58] and CapCodes [13], require at least a footprint of 40×40 mm and 31×21 mm and can, at these sizes, only distinguish eight to 12 individual tangibles.

In contrast, Itsy-Bits distinguishes up to 30 tangibles (10 shapes in three sizes) with smaller footprints of 12×12 mm to 20×20 mm. This size is equal or even smaller than an average fingertip (16-20 mm [9]). If more than 30 tangibles need to be differentiated, n shapes can be fused into a single tangible, resulting in approx. 30^n distinguishable objects (e.g., already 900 for combining two shapes).

3 ITSY-BITS

In this section, we present the operation principle, the composition, and the creation of an Itsy-Bit.

3.1 Terminology

The majority of touchscreens incorporate mutual capacitive sensing that comprises of spatially separated electrodes, arranged as rows and columns [4, 68]. That is, the touch controller measures changes in coupling capacitance between two orthogonal electrodes [10] which forms a *capacitive image* (see Figure 5). Consequently, we apply imaging terminology when appropriate. For instance, a *pixel* of such a capacitive image represents the differences in electrical capacitance (in Picofarad) between the baseline measurement and the current measurement at the corresponding intersection.

Since touch controllers perform finger tracking with a modest resolution (as big as 4x4mm for a single pixel), such capacitive images are of very low resolution (i.e., only 6 PPI compared to

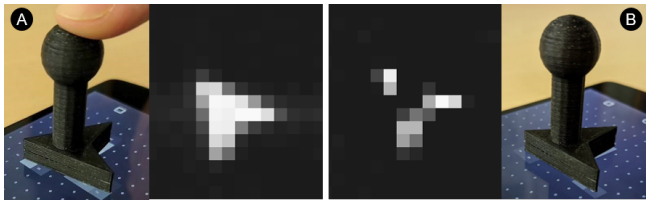

Figure 3: The difference in capacitive images between a touched (A) and untouched (B) conductive shape. The mere presence of untouched objects remains detectable by analyzing the still active pixels.

445 PPI of a display on the same device). As a result, the image of a conductive shape is considerably blurred despite the high spatial resolution of the shape itself (see Figure 2).

3.2 Operation Principle

Based on capacitive sensing, Rekimoto [45] first proposed to embed conductive materials into tangibles to detect them on capacitive touchscreens. Following this idea, an Itsy-Bit contains a conductor that reaches from the location where it is gripped to the location where a conductive shape, placed at the bottom side, contacts the capacitive touchscreen. When the user touches the tangible, the conductor capacitively couples the finger – through the conductive shape – to the touchscreen. This results in a detectable change in capacitances at the pixels that are covered by the conductive shape and, hence, in a capacitive image of the corresponding shape.

Following this general principle, Itsy-Bits employs a machine-learning approach that operates as follows:

(1) Capacitive images are constantly searched for blobs of lit pixels above a minimal threshold (see Figure 2B).
(2) These blobs are then fed to a machine learning model that classifies all shapes, their size, and orientation (see Figure 2C). We detail the model in Section 5.
(3) If a blob is classified as a known shape, the blob's position is tracked as the position of the tangible.

In addition, Itsy-Bits is able to detect untouched tangibles through a weaker capacitive signal since the filled conductive shape connects adjacent pixels (which affects the mutual capacitance). While this signal is too weak for classification, it remained in our informal tests for over two hours and is sufficient to determine whether the tangible still rests untouched on the touchscreen by analyzing whether the pixels in the corresponding blob (see Figure 3B) are above a noise threshold that we defined based on an average of recorded noise images.

3.3 Composition

An Itsy-Bit is a 3D-printed material composite, which consists of the following structures (see Figure 2):

(1) The *grip structure* needs to be touched by the user to eventually form a path between the user and the touchscreen.
(2) The *shape structure* is embedded at the outside of an object and, in case of interaction, touches the screen. Its shape and size differ for each tangible.

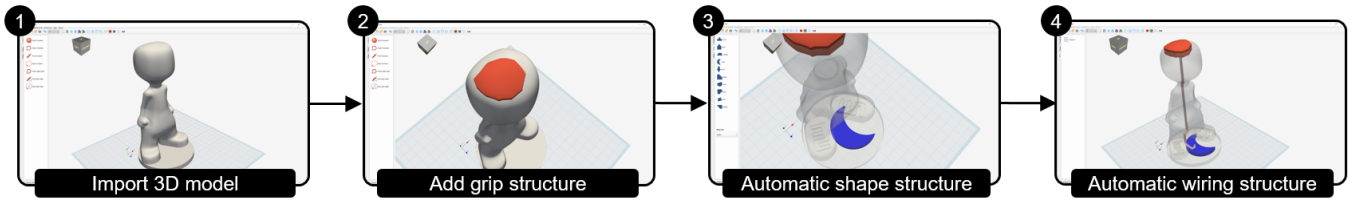

Figure 4: Using a design tool, non-expert users (1) import a 3D-printable model and add (1) grip and (2) shape structures. The tool then automatically embeds necessary wires (4) and provides an export of 3D models for multi-material printing.

(3) The *wiring structure* connects the grip structure to the shape structure so that the user is connected to the touchscreen when gripping the tangible.

All structures are made of a conductive polymer and can have a custom size and 3D shape. Multiple grip, wiring, and shape structure can be embedded within a 3D-printed object.

3.4 Fabrication and Interaction

To create a tangible user interface with a set of individualized Itsy-Bits, a creator performs three steps – design, print, interact – as described in the following:

3.4.1 Design. To facilitate the creation and use of Itsy-Bits also for non-expert users in 3D modeling, we propose a graphical design tool. As depicted in Figure 4, a creator can fabricate a tangible consisting of insulating and conductive material as follows:

(1) The creator imports an arbitrary volumetric 3D model (obtained, for instance, via 3D scanning or an online platform for 3D models) into the design tool which is then displayed in a standard 3D view.
(2) The creator adds custom-shaped grip structures by using different tools, such as lasso or polyline selection of subsurfaces of the object.
(3) The creator then adds one or multiple shape structures to the bottom of the tangible (or any other desired location that may touch the screen). To that end, the creator selects the respective shape and size from a sidebar and clicks on an appropriate 3D position on the model. For further support, the design tool alternatively provides automatic recognition of a surface of a suitable size on the bottom of the object.
(4) When the creator has finished designing, the design tool creates wiring structures between each conductive shape and all grip structures. The auto wiring uses A* operating on a voxelized representation of the 3D model (cf. [49, 51]). On export, the tool creates 3D models for all insulating and conductive structures required for 3D printing using Boolean operations known from constructive solid geometry [37].

3.4.2 Print. Using the exported 3D models, the creator fabricates all Itsy-Bits using a commodity multi-material 3D printer in a single pass (we detail on the fabrication details in Section 4.3). After printing, the tangible can be used directly without further assembly.

3.4.3 Interact. Itsy-Bits directly integrates with standard application development on mobile devices: The creator uses their normal workflow to create a standard touch-enabled application. For tangible interaction, they can receive events containing the shape, size, orientation, and location of an Itsy-Bit when it is placed, moved, or removed from the touchscreen.

The machine-learning model that recognizes Itsy-Bits can be deployed and executed in real-time on standard mobile devices capable of accessing capacitive images. While a single classification requires only milliseconds on an average-class smartphone (e.g., approx. 30 ms on a Nexus 5 used for prototyping), the position of the tangibles is estimated via standard blob tracking to further improve performance. For the same reason, the orientation is only computed on request by the application.

4 GROUND TRUTH DATA COLLECTION

As illustrated in Figure 5, standard shape recognition (e.g., counting defects in the convex hull) is hardly applicable to the low resolution of capacitive images. Therefore, we investigate the feasibility of different supervised machine learning approaches. We conducted a data collection study using optical tracking (as in [30]) to gain ground truth data for shape identification and rotation estimation.

4.1 Selection of Shapes

4.1.1 Types of Shapes. Intuitively, distinguishing conductive shapes on a low-resolution capacitive image requires a set of *maximally* distinct shapes. To find a suitable algorithm to generate shapes with such properties, we conducted a literature review (e.g., [2, 44, 55, 67]). However, to the best of our knowledge, there is no algorithm that produces a (minimum) set of maximally distinguishable shapes.

Therefore, we have chosen an alternative approach that follows the hypothesis that shapes with a specific name in our language have a high visual distinctiveness due to which they were uniquely named. Hence, we considered the set of named shapes, many of which are already known from school, and selected representative examples of the most common shapes: ARROW, CIRCLE, CROSS, HEART, HEXAGON, MOON, PARALLEL, SQUARE, STAR, and TRIANGLE.

As each shape should be rotation-variant, we modified all shapes that did not fulfill this criterion with a linear cut (CIRCLE, CROSS, HEXAGON, PARALLEL, SQUARE, STAR, and TRIANGLE). To avoid confusion, we intentionally left the naming of the original shapes intact (e.g., we still call the *5-edged linear cut square* a square).

Although we focus on named shapes, these already cover a variety of geometric primitives, such as (1) varying curvatures, (2) convex and concave corners and curves, and (3) different number of edges. By varying these geometric features, the shapes cover different amounts of a capacitive pixel and, thus, maximize the

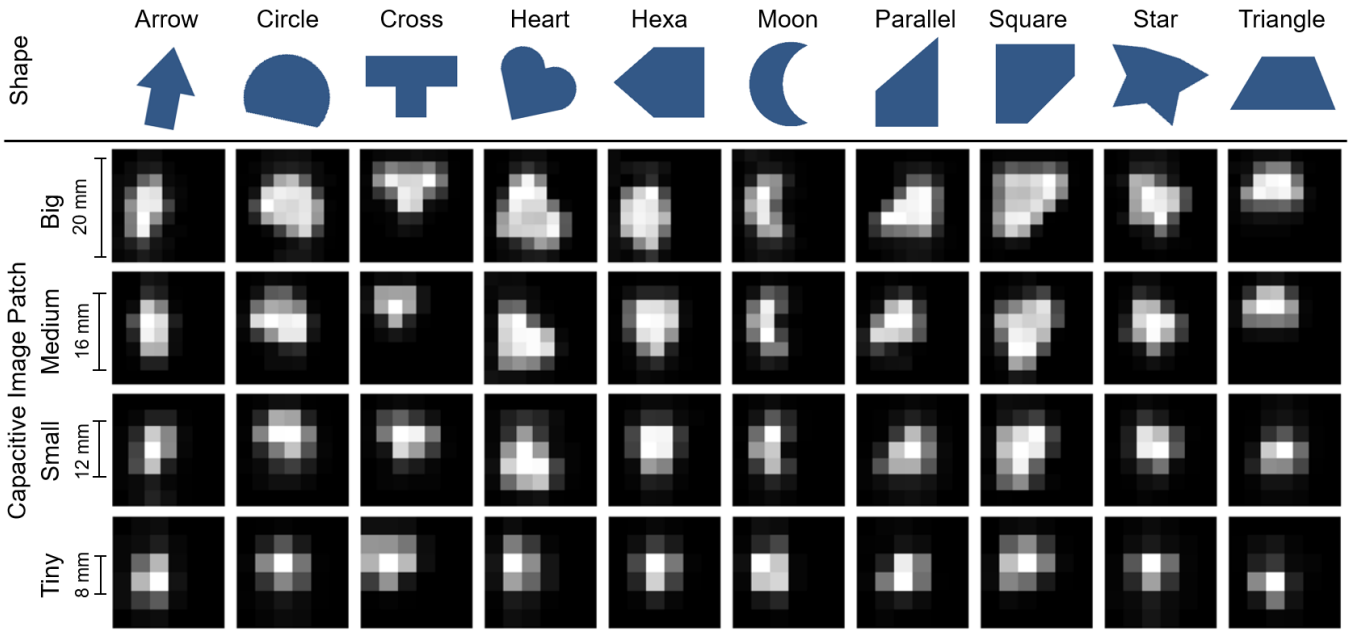

Figure 5: The set of 10 rotation-variant shapes (named after the shape it originated from) and a sample of a corresponding BIG (20×20 mm), MEDIUM (16×16 mm), SMALL (12×12 mm), and TINY (8×8 mm) capacitive image.

chances of being correctly classified by the CNN. As the capacitive images are of very low resolution, the key for differentiation is a set of geometric features that remain when scaled down. This set of shapes is, of course, only one possibility we have chosen in the absence of a suitable generation algorithm.

4.1.2 Sizes of Shapes. Different sizes of shapes are one additional important factor. As one capacitive pixel equals ~4 *mm*, we opted for the following four sizes which multiples of 4 *mm* and normalized to the shape's longest edge: TINY (8×8 *mm* in a 2×2 grid), SMALL (12×12 *mm* in a 3×3 grid), MEDIUM (16×16 *mm* in a 4×4 grid), and BIG (20×20 *mm* in a 5×5 grid). We considered shapes the size of a 1×1 grid, i.e., only a single capacitive pixel, infeasible.

4.2 Apparatus

We used a grounded off-the-shelf LG Nexus 5 device (touch controller Synaptics ClearPad 3350) with a modified kernel to access the capacitive images (without filters or calibration) as described in previous work [16, 29, 31, 39, 53, 65]. Figure 5 depicts exemplary capacitive images. We developed an application that logs the capacitive images at a sampling rate of 20 fps. Of note is, that the sampling rate is only limited due to the fact that we use a debug interface of the touch controller (in line with previous work [16, 29, 53]).

To record object movements on the touchscreen with an accuracy of less than one millimeter, we used an optical tracking system (OptiTrack with 200 fps) with five cameras. We mounted the cameras at a table (see Figure 6B). To track each tangible, we created an OptiTrack trackable (4 markers) which was attached to the objects. Using the trackable, we were able to track the exact rotation and position of the conductive shape on the touchscreen. To determine the relative position of the tangibles on the screen, we calibrated the upper left corner as the origin.

4.3 Fabrication of Shapes

We printed an object for each of the 10 conductive shapes in all four sizes (in total, 40 objects as depicted in Figure 6A). We added an extra handle for users to make rotating easier, and a screw hole to attach an optical marker for ground-truth tracking.

The objects were 3D-printed on a Prusa MK3 with Multi-Material Upgrade 2 and only commercially available printing material. The conductive structures consist of carbon-doped Proto-Pasta Conductive PLA (volume resistivity of $30 - 115\,\Omega * \text{cm}$). All materials were printed with a 0.4 mm thick nozzle at a temperature of 215 °C and a speed of 18 mm/s. To maximize conductivity, we printed all conductive structures with 100 % infill density.

4.4 Participants

While research frequently utilizes mechanical apparatuses to evaluate capacitive approaches technically (see for example [14, 68]), we opted to record our data set with users to ensure that our model is robust against user-specific capacitive effects and hand characteristics. Therefore, we recruited five participants between the ages of 27 and 32 (mean age 29.2). All participants received an introduction to the system before exploring it freely until they felt comfortable. We have complied with all relevant hygiene and infection control guidelines.

4.5 Design and Procedure

After obtaining informed consent, we briefed the participants on the conductive shapes and the data collection procedure. We instructed

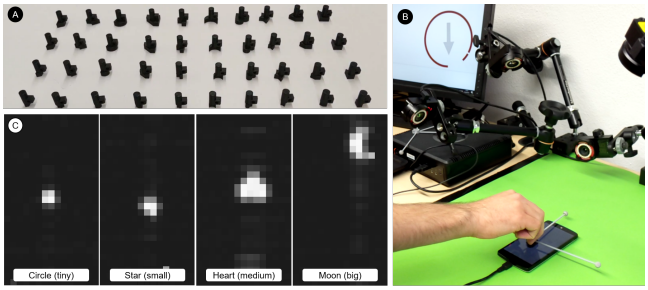

Figure 6: For the data collection study, we printed test objects of all conductive shapes and sizes (A) and recorded their position and rotation on the touchscreen with optical tracking (B) together with the corresponding capacitive images (C).

the participants to rotate all 40 objects as follows: A monitor next to the participant visualized a 2D circle and an arrow which represented the real-time optical tracking data (see Figure 6B). This application ensured that each orientation (1° bins) was hit at least once by providing live feedback. Also, the system recorded further instances of an orientation to gain as much training data as possible. Erroneous data recording was prohibited as the recording stopped as soon as the trackable left the surface of the mobile device. We instructed participants to move and rotate each object slowly due to the lower sampling rate of the screen and also vary the position of the object on the screen. A session lasted 80 minutes per participant on average.

4.6 Results and Post-Processing

In total, we collected a data set that matches 269,867 capacitive images (per user μ = 53,973.4, σ = 5346.3) to their originating conductive shape, ground-truth orientation, and position (see Figure 6C for a small set of samples). To that end, we merged the capacitive images with the optical tracking data using timestamps. Before each study run, we synchronized the time of the Nexus 5 and the optical tracking to a local time server (maximal offset < 0.05s).

5 RECOGNITION OF CONDUCTIVE SHAPES ON CAPACITIVE IMAGES

Based on our data set of low-resolution capacitive images mapped to ground-truth position and orientation, we have trained and validated different models for shape and size classification as well as rotation estimation.

5.1 Pre-Processing

To train a position-invariant model, we first used OpenCV's contour detector to identify the blob of pixels. We then cropped the blob into a new empty capacitive image onto the upper left corner. This approach enables to identify multiple tangibles placed on the touchscreen simultaneously. We removed all other capacitive images which do not contain exactly one touch blob (e.g., a tangible was lifted, or a finger was additionally touching the display).

After this step, we have 193,145 capacitive images. We use a training-test-validation split of 60 % : 20 % : 20 %, which results in 115,887 training, 38,629 test, and 38,629 validation samples. We split

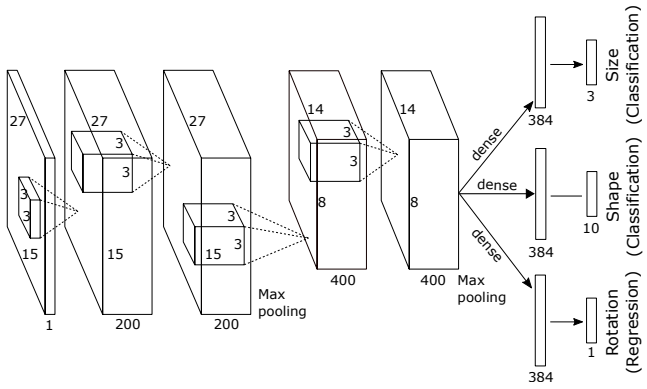

Figure 7: CNN architecture for the model that classifies shapes and sizes, and estimates rotation.

the data set based on the timestamps (1) to split participant-wise (as studies ran subsequently) and (2) to avoid including similar samples (i.e., subsequent images of the same shape) in different data sets. Using this approach, we ensure that we generalize beyond users by approx. splitting subjects 1-3 to training, 4 to test and 5 to validation.

5.2 Modeling

Our modeling process consists of two phases. First, we have experimented with heuristics and basic machine learning techniques to show that recognizing shapes and their attributes is not a trivial challenge on low-resolution images. In the second step, we have applied state-of-the-art deep learning techniques and show that they are superior with adequate performance.

5.2.1 Heuristics and Basic Machine Learning. We trained a support-vector machine (SVM) and a kNN with pixel count and sum/mean/std of the image capacitance as the feature vector. The SVM achieved an accuracy of 17.5% which outperforms the 16.9% by the kNN model. Using the raw blob data as input, the random forest achieved a promising accuracy of 58.3%. We performed a grid search to find optimal hyperparameters for these models and excluded TINY shapes in all basic models as these shapes were even more infeasible to classify.

While these models are lightweight, the accuracy is not sufficient and makes a rotation estimation on top infeasible. Beyond others, two of the main reasons are: (1) the low resolution of the capacitive image which makes an object recognition task a highly non-linear challenge which is hardly solvable with simple models, and (2) the insufficient difference in pixel count between all shapes throughout all sizes, making it an undesirable measure for classifying shape and size at the same time. A possible addition to this initial experiment could be feature extraction using computer vision techniques. However, the richness of the input signal would still be the same while unnecessary complexity is added compared to a pure deep learning technique.

5.2.2 Convolutional Neural Networks. Since convolutional neural networks (CNNs) are recently the state-of-the-art algorithm for

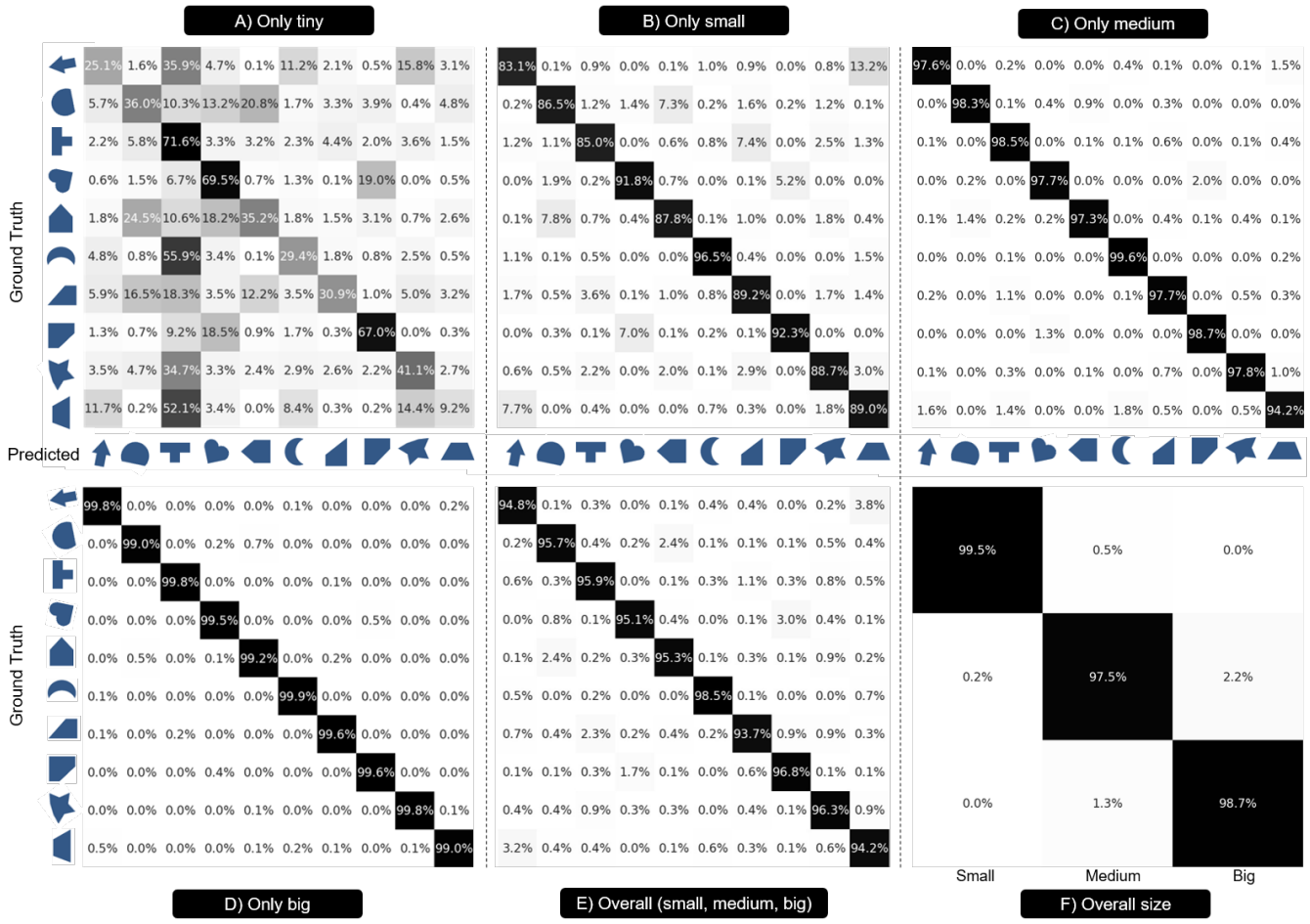

Figure 8: Confusion matrices for the classifiers of shapes of a single size only (A-D). Also, confusion matrices for the combined model (including SMALL, MEDIUM, and BIG) classifying both shape (E) and size (F).

image data, we implemented a CNN with Keras based on the TensorFlow backend. To find the hyperparameters that perform the best on the test set, we performed a grid search as proposed by Hsu et al. [18]. If we do not report a hyperparameter, we applied the standard value (e.g., optimizer settings) as reported in Keras' documentation.

Figure 7 shows the architecture of our CNN that classifies the shape and size of the tangible and estimates its rotation. The input consists of a $15 \times 27\,px$ capacitive image, which is first processed by two groups of convolution and pooling layers. Each convolution layer has a 3×3 kernel while pooling was implemented as a 2×2 max pooling. The result of the convolution layers is then passed on to three branches: a shape and a size identification branch, as well as a rotation estimation branch. The shape and size identification branches are a neural network for classification which uses a softmax activation function in the output layer and categorical cross-entropy as its loss function. The rotation estimation branch solves a regression problem with a linear activation function in the output layer and a root-mean-squared-error of the minimum angular error as its loss function: $loss = atan_2(sin(y_{true} - y_{pred}), cos(y_{true} - y_{pred}))$. We used an Adam optimizer with an initial learning rate of 0.001. Further, we applied batch normalization and a dropout of 0.4 after each max pooling and hidden dense layer to counteract overfitting.

5.3 Model Training and Validation

While we used the training and test set for model development, we use the validation set to calculate the following accuracies. Figure 8 shows the confusion matrices for all classifiers.

5.3.1 Models for a Single Size. First, we trained four different models per size to assess their feasibility for identification and rotation estimation. Our results show that TINY shapes cannot be accurately identified with a shape accuracy of only 50.15 % and a rotation error of 44.93 °. All other shapes can be recognized with the following shape accuracy (SA) and rotation error (RE): (1) for size SMALL: SA 88.86 % and RE 9.05 °, (2) for size MEDIUM: SA 97.77 % and RE 7.27 °, (3) for size BIG: SA 99.51 % and RE 5.96 °.

For the SMALL shapes, the model achieved the lowest accuracy. While still promising, the convolution layers most probably cannot

extract enough signal from the raw data for the dense layers to learn both shape and rotation because of the decreasing shape size in combination with the very low resolution.

5.3.2 Combined Model. Second, we trained a combined model for SMALL, MEDIUM, and BIG shapes with the aforementioned size branch. The model achieves a shape accuracy of 95.63 %, a size accuracy of 98.57 %, and a rotation error of 6.53 ° averaged over all shapes. When validating the combined model with only one size, it achieves a shape accuracy of 89.93 % for SMALL, 97.56 % for MEDIUM, and 99.46 % for BIG.

5.3.3 Models with a Subset of Shapes. Using only a subset of shapes further increases the accuracy. To obtain a subset, shapes with the highest sum of misclassifications (i.e., per row) in the confusion matrix of the combined model should be subsequently removed. To illustrate the potential of shape subset models, we trained two models for only size SMALL with five shapes (CROSS, MOON, SQUARE, CIRCLE, ARROW) with 96.97 %, and six shapes (CROSS, MOON, SQUARE, CIRCLE, ARROW, STAR) with 95.58 % accuracy.

5.4 Cross-Device Validation

In addition to the established validation procedure, we have investigated how our model, trained with data obtained from a touchscreen of a smartphone (4.95"), performs on a differently-sized touchscreen. Furthermore, we were interested in the question whether an app-controlled recording method, a more viable procedure than using expensive optical tracking systems, is suitable to obtain additional raw data (e.g., for additional conductive shapes).

5.4.1 Procedure. We recorded capacitive images without an optical tracking system as follows: Using a mobile application, users follow simple instructions on how to move and rotate the tangible on the display (see Figure 9A). To this end, the application displays a visual representation of a rotated shape in the correct size and instructs the user to exactly cover it with the object. After placing and holding the object still, the application records 2 seconds (approx. 14 samples obtained from the touchscreen's debug interface). The user is then instructed to repeat this procedure for all of the conductive shapes, sizes, and rotations with configurable step-size.

As a proof-of-concept, we carried out the procedure on a tablet (Samsung Galaxy Tab S2 with 9.7" display, approx. 6 PPI for capacitive images, touch controller Synaptics S500B) with a single user (different from the participants of the collection study). In total, we recorded 14.824 images (for all 10 shapes in 10° steps of rotation and the sizes SMALL, MEDIUM, and BIG, as TINY is infeasible to classify). This process takes around one minute to record all rotations of a single conductive shape (hence approx. 30 minutes for 10 shapes in three sizes). As our models operate on capacitive images with size $15 \times 27\ px$, we cut out an image of this size from all tablet images (sized $37 \times 49\ px$) that covered the recorded shape.

5.4.2 Results. We used the recorded data as an additional validation set for the aforementioned combined shape and size classifier. Figure 9 shows the corresponding confusion matrices for shape (B) and size (C). Our results show that shapes can be identified with a high accuracy of 93.34 %. The rotation error of 17.87 ° is increased compared to the original validation (6.53 °). While this

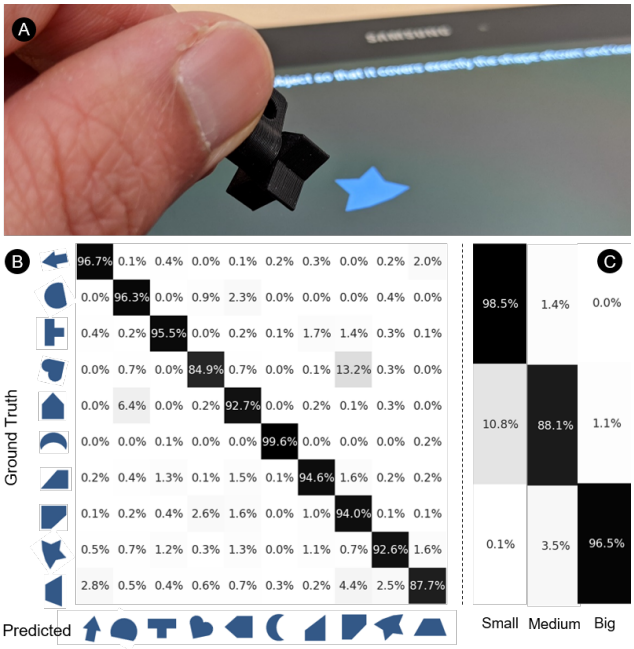

Figure 9: (A) The application to record capacitive images without optical tracking. Also, the confusion matrices for the cross-device validation of the combined shape (B) and size (C) classifier.

could be due to the change in device, this could also be due to the manual recording procedure (i.e., users are probably not able to input a target rotation as precise as provided by optical tracking). This is an interesting aspect that should be further addressed by a study combining manual user input with optical tracking. Moreover, the size accuracy of 90.36 % is lower than the original validation (98.57 %). Interestingly, this mainly affects the MEDIUM size (i.e., the accuracy for SMALL and BIG is 97.5 %) as there are more possibilities for confusion (i.e., both BIG and SMALL can overlap with MEDIUM).

5.5 Mobile Performance

We froze and exported the trained model into a protocol buffer file which we run with *TensorFlow Lite* on both devices (Nexus 5 and Samsung Galaxy Tab S2). A model inference takes 29.2 *ms* (min = 19 *ms*, max = 61 *ms*, SD = 6.01 *ms*) averaged over 1000 runs. This is faster than the 20 fps sampling rate for the capacitive images, which even enables the model to run in the background continuously. Moreover, we reduce load induces due to continuous model inferences by tracking each blob after an initial classification using a soft majority voting (first five frames). The model inference time can be further reduced with model optimization [1, 15], and recent smartphones optimized for neural networks.

6 EXAMPLE APPLICATIONS

In the following, we present example applications for Itsy-Bits that illustrate the broad applicability of tangibles with a small footprint.

Figure 10: Itsy-Bits enhances editing on small devices involving frequent tool changes (A), makes interactive board games more individualized (B), serves as a mobile mixing console (C), or improves tangible learning experience (D).

6.1 Tangible Editing Experience

While mobile work has become increasingly popular and widespread, digital editing is, especially on small form-factors, often cumbersome as many editing functions need to be frequently switched which usually requires many consecutive touch inputs. Therefore, our prototype (inspired by [11]) maps different tangibles to specific editing functions, utilizing the familiarity of objects known from analog editing to provide intuitive mappings. For example, a 3D-printed pen can be used to highlight text while an eraser removes it (see Figure 10A). This prototype also offers a comment bubble, which triggers a comment at the detected position, and a stamp known from paper works, which inserts a symbolic imprint.

In general, such editing interactions are very versatile and generalize not only to the exemplary document editor but also, for instance, to painting, planning, or 3D modeling applications. The strength of Itsy-Bits lies in (1) its small footprint, which allows for precise input, and (2) the versatility of 3D printing and its strong adaptability to a variety of scenarios and user needs.

6.2 Interactive Board Games

This example presents a board game, in which players can compete using different Itsy-Bits: a ship, a treasure chest, and a lighthouse (see Figure 10B). The first player uses the tangible treasure chest to sink a virtual counterpart into the sea by tapping it on the screen. The second player can rotate the lighthouse to reveal the treasure on the ocean floor using the emitted virtual light beam. Once found, the player can lift the treasure using the ship tangible.

This application example illustrates the power of Itsy-Bits to generate a variety of new tangible user experiences that are no longer limited to mass production, but also enable highly individualized game characters (e.g., 3D scanned miniature versions of the players) on the most widespread form factors of capacitive touchscreens.

6.3 Mobile Mixing Console

Mixing consoles require knobs and sliders for precise input but are often large and unportable. While mobile devices are smaller, touch input is often too imprecise, mainly due to the fat-finger problem and the lack of tangibility. With Itsy-Bits, users can print tangibles, similarly shaped to the corresponding instrument, to control their volume, balance, and highs (inspired by [23]). For instance, moving a guitar-shaped object vertically on the touchscreen changes its volume, while rotating adjusts the balance (see Figure 10C), enabling direct and frequent switching between instruments.

This example is representative for mobile scenarios that are profitably on the go where smaller form factors (such as smartphones) are more convenient. In these cases, Itsy-Bits are small enough to allow screen content (e.g., the absolute volume) to be always visible and also provide the benefits of a tangible user interface.

6.4 Tangible Learning

Tangible interfaces offer the potential for higher technology acceptance by the elderly [56]. In this use case, we leverage the physicality and tangibility of Itsy-Bits for an easy and intuitive implementation of the *Montreal Cognitive Assessment* (MoCA) test, a screening tool for cognitive impairments [42]. In this test, subjects must remember items and replicate them in order. As abstract visualizations on paper is often too challenging, this example (see Figure 10D) allows users to interact with 3D representations of the items (a person, a TV, a camera and, symbols for women and man) on smaller devices.

The last example illustrates the meaning of Itsy-Bits also for specialized user groups and the benefits of multiple tangibles that fit on small devices (e.g., all five objects fit on a single 4.95" smartphone) and are highly individualizable (e.g., the learning performance may be improved when using 3D-printed representations that reassemble familiar objects, animals, or people known from the past).

7 DISCUSSION AND LIMITATIONS

While we present an approach to 3D print small tangible objects that can be recognized on off-the-shelf capacitive touchscreens, our approach has limitations that must be considered.

7.1 Further Interactions

Itsy-Bits detects untouched tangibles if they are touched when placed, moved, or removed from the touchscreen. However, there are undoubtedly cases that our approach does not cover and remain challenging (e.g., throwing tangibles onto the touchscreen or a movement induced by tilting the touchscreen).

Moreover, the maximal rotation speed is currently limited by the sampling rate of the debug interface (approx. 20 fps). However, recent touch controllers already feature faster sensing (e.g., 150Hz with the Synaptics S7817) that will improve the performance.

Since this paper focuses on the fundamental properties of CNN shape recognition (with rotation and size), we did not consider finger touches.

As a proof-of-concept, we trained a binary CNN classifier for finger versus shape with an accuracy of 99.62 % based on the open data set *CapFingerId*[2], which was recorded on the same touchscreen as our ground truth data.

7.2 Cross-Device Validation

While we would have preferred to validate our model with more than two devices, access to capacitive images is still limited on many devices. Nevertheless, we see our cross-device validation as a first step showing the capabilities of our model and outlining a procedure to validate future devices without optical tracking.

7.3 Scalability

The number of distinguishable tangibles can be scaled up in two ways: First, combining n shapes scales with 10^n (shapes of only one size) or 30^n (shapes of three sizes), but requires a gap of two pixels between individual shapes (e.g., four SMALL shapes would be sized 26×26 mm, which is still smaller than many related works, and already encode up to 10^4 IDs).

Second, additional shapes can be created based on the presented set by adding edges or cutting inner parts. As a heuristic, a shape should have a sufficient difference not only in high resolution but also when heavily scaled down. This heuristic also explains why some of the shapes are less distinguishable when downsized: For example, CIRCLE and HEXA are interchanged in 2.4% (see Figure 8E, cell [2,5]) as the rounding of the CIRCLE and the tip of the HEXA are blurred in low resolution and, thus, produce similar images. The further the resolution is reduced, the greater the effect, which also explains the effect of the shape's size on the accuracy.

7.4 Data Collection Without Optical Tracking

We opted for a controlled setup to collect our training data, as we otherwise could not guarantee a proper data set, especially for rotation estimation. However, training data for the rotation estimation might also be generated by using simple image rotation algorithms or via an application that visually guides end-users in proper placing, moving, and rotating a new tangible. As a proof-of-concept for the latter, we successfully validated a data set that was manually obtained on another device only via an application.

7.5 Improving the Recognition Accuracy

While we already show a usable classification accuracy and rotation estimation, both could be improved with different methods. A higher touch-sensing resolution helps especially in recognition and rotation estimation as the edges become more visible. As our cross-device validation has shown, the current model can already be transferred to another form factor without further changes.

Moreover, more specialized deep learning models could be used to improve accuracy in general. Techniques such as ensembles, bagging, and boosting could consider multiple images or classifiers to improve the model accuracy. With advances in machine learning research, accuracy is likely to improve even further in the future.

7.6 Differences Between Users and Prints

While a single user can perform the Itsy-Bits pipeline, we collected data from multiple users. This enables us to test for differences in recognition accuracy between users. With our data set split, our results indicate that our approach works independent from users, respectively their hand properties.

Also, multiple 3D prints of a conductive shape may vary slightly within the sub-millimeter tolerances of a 3D printer. We printed multiple instances for our evaluation and applications and observed no effect on recognition, most likely because the model generalizes to the high-level shape instead of tiny variations.

8 CONCLUSION

We have presented Itsy-Bits, a fabrication and recognition approach for small-sized tangibles on off-the-shelf capacitive touchscreens. Our contributions include a fabrication pipeline for 3D-printing conductive tangibles which can be recognized by capacitive touchscreens. We present a machine model that reliably identifies 10 different shapes in three sizes with an accuracy of 95.63 % for shape, an accuracy of 98.57 % for size, and a rotation error of 6.53 °. We illustrate the versatility of our approach for engaging tangible user interfaces through a set of example applications.

Future work could evaluate our use cases in long-term study settings. For this, we publicly share our data set, 3D models, and our CNN model, which can be readily deployed on commodity mobile devices[3]. This enables researchers to use and extend the recognition pipeline and build exciting user interfaces with tangibles that have a footprint as small as a fingertip.

9 ACKNOWLEDGMENTS

We thank David Reichenbach, Marco Fendrich, Tobias Dörlam, and the reviewers for their valuable support. This work has been co-funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 211500647 & 326979514 and by the LOEWE initiative (Hesse, Germany) within the emergenCITY center.

[2] github.com/interactionlab/CapFingerId
[3] github.com/telecooperation/itsy-bits

REFERENCES

[1] Sajid Anwar, Kyuyeon Hwang, and Wonyong Sung. 2017. Structured Pruning of Deep Convolutional Neural Networks. *J. Emerg. Technol. Comput. Syst.* 13, 3, Article 32 (Feb. 2017), 18 pages. https://doi.org/10.1145/3005348

[2] Thomas Auer and Martin Held. 1996. Heuristics for the Generation of Random Polygons. In *Proceedings of the 8th Canadian Conference on Computational Geometry*. Carleton University Press, 38–43. https://doi.org/10.5555/648249.751880

[3] Moritz Bächer, Benjamin Hepp, Fabrizio Pece, Paul G. Kry, Bernd Bickel, Bernhard Thomaszewski, and Otmar Hilliges. 2016. DefSense: Computational Design of Customized Deformable Input Devices. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 3806–3816. https://doi.org/10.1145/2858036.2858354

[4] Gary Barrett and Ryomei Omote. 2010. Projected-capacitive touch technology. *Information Display* 26, 3 (2010), 16–21. http://large.stanford.edu/courses/2012/ph250/lee2/docs/art6.pdf

[5] Patrick Baudisch, Torsten Becker, and Frederik Rudeck. 2010. Lumino: Tangible Blocks for Tabletop Computers Based on Glass Fiber Bundles. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems - CHI '10*. ACM Press, New York, New York, USA, 1165. https://doi.org/10.1145/1753326.1753500

[6] Eric Brockmeyer, Ivan Poupyrev, and Scott Hudson. 2013. PAPILLON: Designing Curved Display Surfaces with Printed Optics. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology - UIST '13*. ACM Press, New York, New York, USA, 457–462. https://doi.org/10.1145/2501988.2502027

[7] 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. https://doi.org/10.1007/978-3-319-22701-6

[8] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. 2012. CapStones and ZebraWidgets: Sensing Stacks of Building Blocks, Dials and Sliders on Capacitive Touch Screens. In *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems - CHI '12*. ACM Press, New York, New York, USA, 2189. https://doi.org/10.1145/2207676.2208371

[9] Kiran Dandekar, Balasundar I Raju, and Mandayam a Srinivasan. 2003. 3-D Finite-Element Models of Human and Monkey Fingertips to Investigate the Mechanics of Tactile Sense. *Journal of Biomechanical Engineering* 125, 5 (2003), 682. https://doi.org/10.1115/1.1613673

[10] Li Du. 2016. An Overview of Mobile Capacitive Touch Technologies Trends. *arXiv preprint arXiv:1612.08227* (2016), 4 pages.

[11] Lisa A. Elkin, Jean-Baptiste Beau, Géry Casiez, and Daniel Vogel. 2020. Manipulation, Learning, and Recall with Tangible Pen-Like Input. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376772

[12] Hyunjae Gil, DoYoung Lee, Seunggyu Im, and Ian Oakley. 2017. TriTap: Identifying Finger Touches on Smartwatches. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. ACM, New York, NY, USA, 3879–3890. https://doi.org/10.1145/3025453.3025561

[13] Timo Götzelmann and Christopher Althaus. 2016. TouchSurfaceModels: Capacitive Sensing Objects Through 3D Printers. In *Proceedings of the 9th ACM International Conference on PErvasive Technologies Related to Assistive Environments (PETRA '16)*. ACM, New York, NY, USA, 22:1–22:8. https://doi.org/10.1145/2910674.2910690

[14] Tobias Alexander Große-Puppendahl. 2015. *Capacitive Sensing and Communication for Ubiquitous Interaction and Environmental Perception.* Ph.D. Dissertation. Technische Universität Darmstadt, Darmstadt.

[15] Song Han, Huizi Mao, and William J. Dally. 2015. Deep Compression: Compressing Deep Neural Network with Pruning, Trained Quantization and Huffman Coding. *CoRR* abs/1510.00149 (2015), 14 pages. arXiv:1510.00149 http://arxiv.org/abs/1510.00149

[16] Christian Holz, Senaka Buthpitiya, and Marius Knaust. 2015. Bodyprint: Biometric User Identification on Mobile Devices Using the Capacitive Touchscreen to Scan Body Parts. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. ACM, New York, NY, USA, 3011–3014. https://doi.org/10.1145/2702123.2702518

[17] Jonathan Hook, Thomas Nappey, Steve Hodges, Peter Wright, and Patrick Olivier. 2014. Making 3D Printed Objects Interactive Using Wireless Accelerometers. In *Proceedings of the Extended Abstracts of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 1435–1440. https://doi.org/10.1145/2559206.2581137

[18] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. 2003. A practical guide to support vector classification. (2003), 16 pages.

[19] Sungjae Hwang, Myungwook Ahn, and Kwangyun Wohn. 2013. Magnetic Marionette: Magnetically Driven Elastic Controller on Mobile Device. In *Proceedings of the Companion Publication of the 2013 International Conference on Intelligent User Interfaces Companion (IUI '13 Companion)*. ACM, New York, NY, USA, 75–76. https://doi.org/10.1145/2451176.2451207

[20] Kohei Ikeda and Koji Tsukada. 2015. CapacitiveMarker: Novel Interaction Method Using Visual Marker Integrated with Conductive Pattern. In *Proceedings of the 6th Augmented Human International Conference (AH '15)*. ACM, New York, NY, USA, 225–226. https://doi.org/10.1145/2735711.2735783

[21] Kaori Ikematsu and Itiro Siio. 2018. Ohmic-Touch: Extending Touch Interaction by Indirect Touch Through Resistive Objects. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 521:1–521:8. https://doi.org/10.1145/3173574.3174095

[22] Yoshio Ishiguro and Ivan Poupyrev. 2014. 3D Printed Interactive Speakers. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems - CHI '14*. ACM Press, New York, New York, USA, 1733–1742. https://doi.org/10.1145/2556288.2557046

[23] Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. 2007. The reacTable: Exploring the Synergy between Live Music Performance and Tabletop Tangible Interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. Association for Computing Machinery, New York, NY, USA, 139–146. https://doi.org/10.1145/1226969.1226998

[24] Kunihiro Kato and Homei Miyashita. 2015. ExtensionSticker: A Proposal for a Striped Pattern Sticker to Extend Touch Interfaces and Its Assessment. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, Vol. 1. ACM Press, New York, New York, USA, 1851–1854. https://doi.org/10.1145/2702123.2702500

[25] Kunihiro Kato and Homei Miyashita. 2016. 3D Printed Physical Interfaces That Can Extend Touch Devices. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16 Adjunct*. ACM Press, New York, New York, USA, 47–49. https://doi.org/10.1145/2984751.2985700

[26] Sven Kratz, Tilo Westermann, Michael Rohs, and Georg Essl. 2011. CapWidgets: Tangible Widgets versus Multi-Touch Controls on Mobile Devices. In *Proceedings of the 2011 Annual Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '11*. ACM Press, New York, New York, USA, 1351. https://doi.org/10.1145/1979742.1979773

[27] Han-Chih Kuo, Rong-Hao Liang, Long-Fei Lin, and Bing-Yu Chen. 2016. GaussMarbles: Spherical Magnetic Tangibles for Interacting with Portable Physical Constraints. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 4228–4232. https://doi.org/10.1145/2858036.2858559

[28] Gierad Laput, Eric Brockmeyer, Scott E. Hudson, and Chris Harrison. 2015. Acoustruments: Passive, Acoustically-Driven, Interactive Controls for Handheld Devices. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, New York, New York, USA, 2161–2170. https://doi.org/10.1145/2702123.2702414

[29] Huy Viet Le, Thomas Kosch, Patrick Bader, Sven Mayer, and Niels Henze. 2018. PalmTouch: Using the Palm as an Additional Input Modality on Commodity Smartphones. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3173574.3173934

[30] Huy Viet Le, Sven Mayer, and Niels Henze. 2018. InfiniTouch: Finger-Aware Interaction on Fully Touch Sensitive Smartphones. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3242587.3242605

[31] Huy Viet Le, Sven Mayer, and Niels Henze. 2019. Investigating the Feasibility of Finger Identification on Capacitive Touchscreens using Deep Learning. In *24th International Conference on Intelligent User Interfaces* (Marina del Ray, CA, USA) *(IUI '19)*. ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3301275.3302295

[32] Simon J Leigh, Robert J Bradley, Christopher P Purssell, Duncan R Billson, and David a Hutchins. 2012. A Simple, Low-Cost Conductive Composite Material for 3D Printing of Electronic Sensors. *PLoS ONE* 7, 11 (Nov. 2012), e49365. https://doi.org/10.1371/journal.pone.0049365

[33] Rong-Hao Liang, Liwei Chan, Hung-Yu Tseng, Han-Chih Kuo, Da-Yuan Huang, De-Nian Yang, and Bing-Yu Chen. 2014. GaussBricks: Magnetic Building Blocks for Constructive Tangible Interactions on Portable Displays. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems - CHI '14*. ACM Press, New York, New York, USA, 3153–3162. https://doi.org/10.1145/2556288.2557105

[34] Rong-Hao Liang, Kai-Yin Cheng, Liwei Chan, Chuan-Xhyuan Peng, Mike Y. Chen, Rung-Huei Liang, De-Nian Yang, and Bing-Yu Chen. 2013. GaussBits: Magnetic Tangible Bits for Portable and Occlusion-Free Near-Surface Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press, New York, New York, USA, 1391. https://doi.org/10.1145/2470654.2466185

[35] Rong-Hao Liang, Kai-Yin Cheng, Chao-Huai Su, Chien-Ting Weng, Bing-Yu Chen, and De-Nian Yang. 2012. GaussSense: Attachable Stylus Sensing Using Magnetic Sensor Grid. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. Association for Computing Machinery, New York, NY, USA, 319–326. https://doi.org/10.1145/2380116.2380157

[36] Rong-hao Liang, Han-Chih Kuo, Liwei Chan, De-Nian Yang, and Bing-Yu Chen. 2014. GaussStones : Shielded Magnetic Tangibles for Multi-Token Interactions on Portable Displays. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. ACM Press, New York, New York, USA, 365–372. https://doi.org/10.1145/2642918.2647384

[37] Martti Mantyla. 1988. *Introduction to Solid Modeling*. WH Freeman \& Co.

[38] 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 (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376189

[39] Sven Mayer, Huy Viet Le, and Niels Henze. 2017. Estimating the Finger Orientation on Capacitive Touchscreens Using Convolutional Neural Networks. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces* (Brighton, United Kingdom) *(ISS '17)*. ACM, New York, NY, USA, 220–229. https://doi.org/10.1145/3132272.3134130

[40] Mohammad Faizuddin Mohd Noor, Andrew Ramsay, Stephen Hughes, Simon Rogers, John Williamson, and Roderick Murray-Smith. 2014. 28 Frames Later: Predicting Screen Touches from Back-of-device Grip Changes. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. ACM, New York, NY, USA, 2005–2008. https://doi.org/10.1145/2556288.2557148

[41] Mohammad Faizuddin Mohd Noor, Simon Rogers, and John Williamson. 2016. Detecting Swipe Errors on Touchscreens Using Grip Modulation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (Santa Clara, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 1909–1920. https://doi.org/10.1145/2858036.2858474

[42] Ziad S. Nasreddine, Natalie A. Phillips, Valérie Bédirian, Simon Charbonneau, Victor Whitehead, Isabelle Collin, Jeffrey L. Cummings, and Howard Chertkow. 2005. The Montreal Cognitive Assessment, MoCA: A Brief Screening Tool For Mild Cognitive Impairment. *Journal of the American Geriatrics Society* 53, 4 (2005), 695–699. https://doi.org/10.1111/j.1532-5415.2005.53221.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1532-5415.2005.53221.x

[43] Makoto Ono, Buntarou Shizuki, and Jiro Tanaka. 2013. Touch & Activate: Adding Interactivity to Existing Objects Using Active Acoustic Sensing. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology - UIST '13*. ACM Press, New York, New York, USA, 31–40. https://doi.org/10.1145/2501988.2501989

[44] Joseph O'Rourke, Heather Booth, and Richard Washington. 1987. Connect-the-dots: A new heuristic. *Computer Vision, Graphics, and Image Processing* 39, 2 (1987), 258 – 266. https://doi.org/10.1016/S0734-189X(87)80169-X

[45] Jun Rekimoto. 2002. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems Changing Our World, Changing Ourselves - CHI '02*. ACM Press, New York, New York, USA, 113. https://doi.org/10.1145/503376.503397

[46] Francisco J. Romero-Ramirez, Rafael Muñoz-Salinas, and Rafael Medina-Carnicer. 2018. Speeded up Detection of Squared Fiducial Markers. *Image and Vision Computing* 76 (Aug. 2018), 38–47. https://doi.org/10.1016/j.imavis.2018.05.004

[47] Munehiko Sato, Ivan Poupyrev, and Chris Harrison. 2012. Touché: Enhancing Touch Interaction on Humans, Screens, Liquids, and Everyday Objects. In *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems - CHI '12*. ACM Press, New York, New York, USA, 483. https://doi.org/10.1145/2207676.2207743

[48] Valkyrie Savage, Colin Chang, and Björn Hartmann. 2013. Sauron: Embedded Single-Camera Sensing of Printed Physical User Interfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. ACM Press, New York, New York, USA, 447–456. https://doi.org/10.1145/2501988.2501992

[49] Valkyrie Savage, Ryan Schmidt, Tovi Grossman, George Fitzmaurice, and Björn Hartmann. 2014. A Series of Tubes: Adding Interactivity to 3D Prints Using Internal Pipes. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM Press, New York, New York, USA, 3–12. https://doi.org/10.1145/2642918.2647374

[50] Martin Schmitz, Martin Herbers, Niloofar Dezfuli, 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 (CHI '18)*. ACM, New York, NY, USA, 182:1–182:8. https://doi.org/10.1145/3173574.3173756

[51] 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 on User Interface Software & Technology (UIST '15)*. ACM Press, New York, New York, USA, 253–258. https://doi.org/10.1145/2807442.2807503

[52] Martin Schmitz, Andreas Leister, Niloofar Dezfuli, Jan Riemann, Florian Müller, and Max Mühlhäuser. 2016. Liquido: Embedding Liquids into 3D Printed Objects to Sense Tilting and Motion. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM Press, New York, New York, USA, 2688–2696. https://doi.org/10.1145/2851581.2892275

[53] Martin Schmitz, Jürgen Steimle, Jochen Huber, Niloofar Dezfuli, and Max Mühlhäuser. 2017. Flexibles: Deformation-Aware 3D-Printed Tangibles for Capacitive Touchscreens. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 1001–1014. https://doi.org/10.1145/3025453.3025663

[54] Martin Schmitz, Martin Stitz, Florian Müller, Markus Funk, and Max Mühlhäuser. 2019. ./Trilaterate: A Fabrication Pipeline to Design and 3D Print Hover-, Touch-, and Force-Sensitive Objects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, Glasgow, Scotland Uk, 1–13. https://doi.org/10.1145/3290605.3300684

[55] Maryann Simmons and Carlo H Séquin. 1998. 2d shape decomposition and the automatic generation of hierarchical representations. *International Journal of Shape Modeling* 4, 01n02 (1998), 63–78.

[56] Wolfgang Spreicer. 2011. Tangible Interfaces as a Chance for Higher Technology Acceptance by the Elderly. In *Proceedings of the 12th International Conference on Computer Systems and Technologies* (Vienna, Austria) *(CompSysTech '11)*. Association for Computing Machinery, New York, NY, USA, 311–316. https://doi.org/10.1145/2023607.2023660

[57] Nicolas Villar, Daniel Cletheroe, Greg Saul, Christian Holz, Tim Regan, Oscar Salandin, Misha Sra, Hui-Shyong Yeo, William Field, and Haiyan Zhang. 2018. Project Zanzibar: A Portable and Flexible Tangible Interaction Platform. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 515:1–515:13. https://doi.org/10.1145/3173574.3174089

[58] Simon Voelker, Christian Cherek, Jan Thar, Thorsten Karrer, Christian Thoresen, Kjell Ivar Øvergård, and Jan Borchers. 2015. PERCs: Persistently Trackable Tangibles on Capacitive Multi-Touch Displays. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology - UIST '15*. ACM Press, New York, New York, USA, 351–356. https://doi.org/10.1145/2807442.2807466

[59] Simon Voelker, Kosuke Nakajima, Christian Thoresen, Yuichi Itoh, Kjell Ivar Øvergård, and Jan Borchers. 2013. PUCs: Detecting Transparent, Passive Untouched Capacitive Widgets on Unmodified Multi-Touch Displays. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces - ITS '13*. ACM Press, New York, New York, USA, 101–104. https://doi.org/10.1145/2512349.2512791

[60] Johnty Wang, Nicolas Alessandro, Sidney Fels, and Bob Pritchard. 2011. SQUEEZY: Extending a Multi-Touch Screen with Force Sensing Objects for Controlling Articulatory Synthesis. In *Proceedings of the International Conference on New Interfaces for Musical Expression*. Oslo, Norway, 531–532.

[61] Yuntao Wang, Jianyu Zhou, Hanchuan Li, Tengxiang Zhang, Minxuan Gao, Zhuolin Cheng, Chun Yu, Shwetak Patel, and Yuanchun Shi. 2019. FlexTouch: Enabling Large-Scale Interaction Sensing Beyond Touchscreens Using Flexible and Conductive Materials. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (Sept. 2019), 109:1–109:20. https://doi.org/10.1145/3351267

[62] Malte Weiss, Julie Wagner, Yvonne Jansen, Roger Jennings, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. 2009. SLAP Widgets: Bridging the Gap Between Virtual and Physical Controls on Tabletops. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems - CHI 09*. ACM Press, New York, New York, USA, 481. https://doi.org/10.1145/1518701.1518779

[63] Cary Williams, Xing Dong Yang, Grant Partridge, Joshua Millar-Usiskin, Arkady Major, and Pourang Irani. 2011. TZee: Exploiting the Lighting Properties of Multi-Touch Tabletops for Tangible 3D Interactions. In *Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems - CHI '11*. ACM Press, New York, New York, USA, 1363. https://doi.org/10.1145/1978942.1979143

[64] Karl Willis, Eric Brockmeyer, Scott Hudson, and Ivan Poupyrev. 2012. Printed Optics: 3D Printing of Embedded Optical Elements for Interactive Devices. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology - UIST '12*. ACM Press, New York, New York, USA, 589. https://doi.org/10.1145/2380116.2380190

[65] Robert Xiao, Julia Schwarz, and Chris Harrison. 2015. Estimating 3D Finger Angle on Commodity Touchscreens. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces* (Madeira, Portugal) *(ITS '15)*. ACM, New York, NY, USA, 47–50. https://doi.org/10.1145/2817721.2817737

[66] Neng-Hao Yu, Polly Huang, Yi-ping Hung, Li-Wei Chan, Seng-yong Yong Lau, Sung-Sheng Tsai, I-Chun Hsiao, Dian-Je Tsai, Fang-I Hsiao, Lung-Pan Cheng, and Mike Chen. 2011. TUIC: Enabling Tangible Interaction on Capacitive Multi-Touch Display. In *Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems - CHI '11*. ACM Press, New York, New York, USA, 2995. https://doi.org/10.1145/1978942.1979386

[67] Chong Zhu, Gopalakrishnan Sundaram, Jack Snoeyink, and Joseph SB Mitchell. 1996. Generating random polygons with given vertices. *Computational Geometry* 6, 5 (1996), 277–290.

[68] Thomas G. Zimmerman, Joshua R. Smith, Joseph a. Paradiso, David Allport, and Neil Gershenfeld. 1995. Applying Electric Field Sensing to Human-Computer Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '95*. ACM Press, New York, New York, USA, 280–287. https:

//doi.org/10.1145/223904.223940