CapContact: Super-resolution Contact Areas from Capacitive Touchscreens

Paul Streli and Christian Holz
Department of Computer Science
ETH Zürich, Switzerland
{paul.streli,christian.holz}@inf.ethz.ch

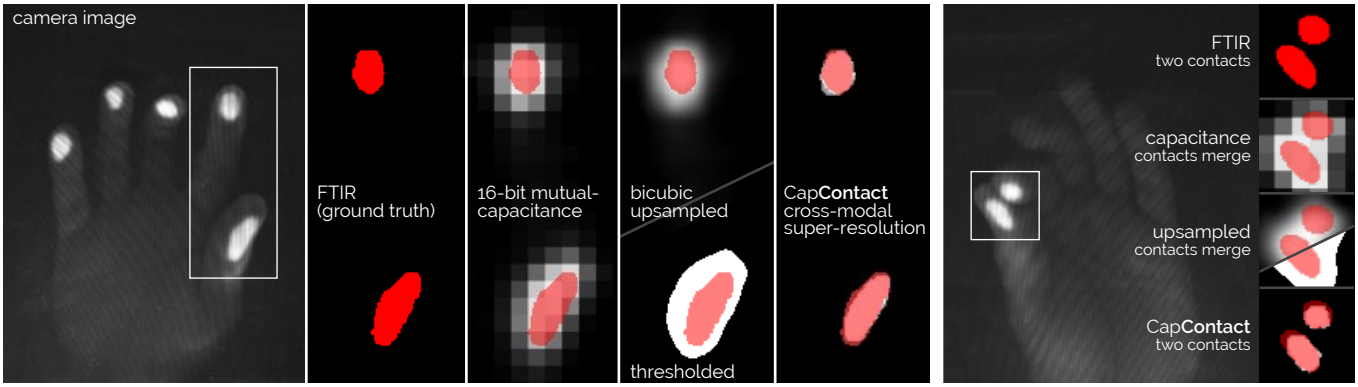

Figure 1: CapContact reconstructs a *high-resolution contact area* between the user's fingers and the touch surface from the typical low-resolution capacitive sensor. We designed and trained an 8× cross-modality super-resolution convolutional neural network on pairs of regular-resolution mutual-capacitance sensor images and ground-truth contact maps from a frustrated total internal reflection setup (FTIR, contact areas overlaid in red in each image). CapContact produces significantly more accurate contact areas than the baseline of thresholded bicubic upsampling on the capacitive images. Unlike existing methods, CapContact can discriminate closely adjacent touches into *separate* contact areas that merge in raw capacitive recordings.

ABSTRACT

Touch input is dominantly detected using mutual-capacitance sensing, which measures the *proximity* of close-by objects that change the electric field between the sensor lines. The exponential drop-off in intensities with growing distance enables software to detect touch events, but does not reveal true *contact areas*. In this paper, we introduce CapContact, a novel method to precisely infer the contact area between the user's finger and the surface from a single capacitive image. At 8× super-resolution, our convolutional neural network generates refined touch masks from 16-bit capacitive images as input, which can even discriminate adjacent touches that are not distinguishable with existing methods. We trained and evaluated our method using supervised learning on data from 10 participants who performed touch gestures. Our capture apparatus integrates optical touch sensing to obtain ground-truth contact through high-resolution frustrated total internal reflection. We compare our method with a baseline using bicubic upsampling as well as the ground truth from FTIR images. We separately evaluate our method's performance in discriminating adjacent touches. CapContact successfully separated closely adjacent touch contacts in 494 of 570 cases (87%) compared to the baseline's 43 of 570 cases (8%). Importantly, we demonstrate that our method accurately performs even at *half of the sensing resolution* at twice the grid-line pitch across the same surface area, challenging the current industry-wide standard of a ~4 mm sensing pitch. We conclude this paper with implications for capacitive touch sensing in general and for touch-input accuracy in particular.

CCS CONCEPTS

• **Human-centered computing** → **Touch screens**.

KEYWORDS

Touch input; Capacitive sensing; Super-resolution; Contact area; Accuracy; Generative adversarial networks;

ACM Reference Format:
Paul Streli and Christian Holz. 2021. CapContact: Super-resolution Contact Areas from Capacitive Touchscreens. In *CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3411764.3445621

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 the author(s) 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.
CHI '21, May 8–13, 2021, Yokohama, Japan
© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-8096-6/21/05…$15.00
https://doi.org/10.1145/3411764.3445621

1 INTRODUCTION

Today's consumer touch devices mostly use a form of capacitive sensing to detect touch input. Most implement mutual-capacitance

sensing, a technology where drive and sense lines are routed orthogonally and digitizers operate in shunt mode [21], measuring the decrease in capacitance between grid lines to detect the proximity of fingers. This technology can scale from curved surfaces on small devices (e.g., Microsoft Band) to wearable and flexible sensors [59] as well as to large tables and wall displays (e.g., Surface Hub)—all using the same sensing approach for interactive purposes [47].

The resolution of touch sensors on interactive devices, i.e., the number of transparent semi-conductor lines stretching across the display, is determined by their primary purpose: reliably detecting (human) touch and, importantly, discriminating adjacent touches. The choice of a standard pitch in this industry, i.e., the spacing between adjacent grid lines, of ~4 mm goes back to the beginning of spatial capacitive sensing [40] and has remained comparable in commercial devices [60]. Although the resolution of lines across a surface may appear low, touch-input locations can be determined with sub-pixel precision as the weighted arithmetic mean of the individual sensor values [47].

Over the past two decades and in parallel to the integration of capacitive sensors in consumer devices (e.g., FingerWorks desktop touchpad [14], smartphones, laptops), research efforts in the human-computer interaction community have uncovered numerous benefits of sensing the touch *contact area*. Starting with explorations on multi-touch tables [3, 25], a variety of contact area-based interaction techniques have since emerged (e.g., [5, 9, 46]). Through these explorations, researchers have recognized the rich set of parameters contained in the shape of a touch [6], culminating in "natural user interfaces" that are operated through intuitive touch, analogous to how we might interact with physical objects in the real world [61].

More recently, researchers have started to investigate the feasibility of deriving touch *shapes* on capacitive touchscreens (e.g., to resolve biometric landmarks such as ears [31] and hands [24], touch shapes, and tangible objects [64]). However, capacitive sensing was never intended to measure touch contact *areas*. Rather, it registers the *proximity* to close-by objects that change the electric field between the grid lines. Sensor values from a digitizer arise from the exponential drop-off in capacitance between the field lines, such that manufacturers can guarantee reliable reports of touch events within 0.5 mm (dubbed "Pre-Touch" [54]) of the protective touch surface above a sensor, while rejecting all those farther away [51]. Through calibration, manufacturers provide a threshold for intensities to reliably determine touch *events* and derive their locations in software; the immediate sensor readings are of limited use to resolve contact *areas*, however.

In this paper, we introduce a method that reconstructs the contact area between the user's finger and the touch surface. Our method *CapContact* comprises a cross-modal training and inference pipeline that, from a single 16-bit capacitive image as input, super-resolves a precise and binary contact mask at 8× higher resolution.

1.1 Reconstructing high-resolution contact *areas* from capacitance sensors

Figure 1 shows an overview of our method. The camera image shows a hand captured from below a transparent surface that integrates mutual-capacitive sensing as well as optical touch sensing. The visible touch contacts result from our frustrated internal reflection setup (FTIR [25]), which illuminates the contact areas between the fingers and the surface. As shown in the four middle columns, inferring the precise contact *area* from capacitive recordings is challenging, which is particularly evident in the raw capacitive sensor values as well as the bicubic upsampled representation. This is because the parts of the fingers that hover just above the surface capacitively couple with comparable intensity and are thus hard to distinguish from actually touching parts. In contrast, our method CapContact reconstructs contact areas that are comparable to those obtained using ground-truth contact from FTIR.

We trained and evaluated CapContact using supervised learning on 16-bit raw capacitive measurements from 10 participants who performed various touch gestures. We compare our results with a baseline of bicubic upsampled intensities as well as the ground-truth shapes obtained from the high-resolution FTIR images.

1.2 The four main benefits of our method

Taken together, we found several benefits for CapContact.

(1) Our method directly benefits current devices by reliably discriminating between adjacent touches, separating them into their precise contact areas as shown in Figure 1 (right), which are not distinguishable using current methods.
(2) CapContact generalizes to even lower-resolution capacitive sensors; even at *half* resolution (i.e., a pitch of 8 mm), CapContact more reliably distinguishes closely adjacent touches than the baseline at *full* resolution—despite never having been fine-tuned on closely collocated touches.
(3) Due to CapContact's accurate reconstruction, derived touch locations are much closer to the center of gravity of the true contact [29, 56] than the center of gravity of the capacitive measurements (Figure 2).
(4) Finally, CapContact delivers the missing piece of information for HCI techniques that were designed for contact area-based interaction to migrate to capacitive touchscreens, including to phones, tablets, and large-screen displays.

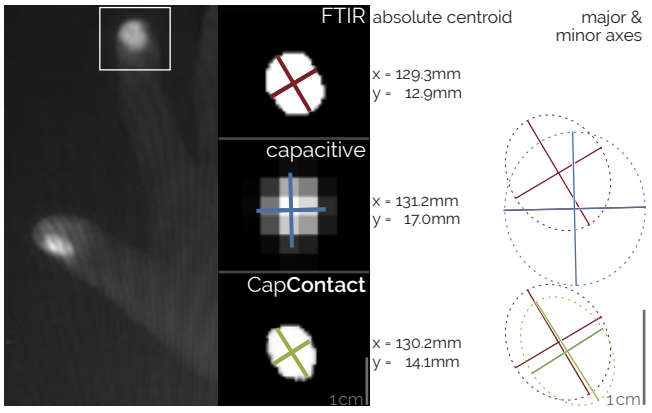

Figure 2: CapContact's high-resolution reconstruction of the contact area between the user's finger and the surface produces a more accurate input location of the touch event compared to the center of mass from the capacitive values.

1.3 Contributions

In this paper, we make the following contributions.

- A 10-participant data collection study that establishes the first corpus of mutual-capacitance sensor images registered with ground-truth contact areas obtained from a frustrated total internal reflection setup.
- CapContact, a generative adversarial network for 8× cross-modality super-resolution. CapContact infers contact sizes and shapes with less than 3% deviation from the ground truth and reduces the contact centroid error by over 20% compared to a baseline of bicubic upsampling.
- A follow-up evaluation of discriminating 4 participants' adjacent touches that merge in capacitive images but are reliably separated by CapContact without any fine-tuning with such cases. A second evaluation on downsampled recordings at *half* resolution verifies that CapContact separates touches at a higher success rate than the baseline at *full* resolution.
- A release of CapContact's implementation, trained models, as well as our collected data corpus to support follow-up research on contact shape-based interaction and context inference on capacitive touchscreens[1].

2 RELATED WORK

Our work is related to capacitive sensing and its use-cases, machine learning on capacitive images, high-precision touch-input, and super-resolution techniques in computer vision.

2.1 Capacitive sensing and use-cases in HCI

Exploring capacitive sensing as an input modality beyond detecting touch locations has a rich history in Human-Computer Interaction. Both, DiamondTouch [11] and SmartSkin [47] pioneered the implementation of the now common sense-line and drive-line pattern for HCI purposes, showing examples of touch separation and tracking, user differentiation, as well as shape recognition (e.g., upsampled palm prints through bicubic interpolation [18, 47] and whole-arm and hand interaction). Grosse-Puppendahl et al. provide an extensive overview of the configurations and use-cases of capacitive sensing in the domain [21]. In here, we focus on efforts that specifically investigated aspects of touch contact, shape, and hover.

Processing capacitive images has been popular for the purpose of extracting additional input information from touch events beyond input location. Reconstructing finger angles has played an important role in this regard, such as in Rogers et al.'s particle filter applied to a low-resolution sensor array [50] or for the purpose of one-handed device operation (e.g., the fat thumb [4]). Complementarily, Wang et al. investigated the feasibility of reconstructing finger angles based on contact area alone using FTIR [55], analyzing the principal components of contact shapes and drawing on temporal observations for better estimates.

Capacitive sensing has also found use for detecting hovering fingers. For example, Rekimoto et al. built capacitive sensing into a numeric pad to establish a preview channel before actual input [48]. More transparently even, Pre-Touch uses an array of self-capacitance sensors inside a touchscreen to unobtrusively detect hover, used finger, grip and input trajectories [27].

Learning from Capacitive Images. Using the developments in the computer vision space, HCI researchers have investigated their suitability for recognizing touch characteristics directly learning from capacitive images. For example, Le et al.'s convolutional neural network (CNN) differentiates between touches of fingers and the palm [37] and Mayer et al.'s CNN estimates finger yaw and pitch directly from the image [42]. An added benefit of these investigations is the increased availability of datasets for future research on capacitive sensor data, in Mayer et al.'s case synchronized with recordings from a high-precision motion capture system. InfiniTouch went a step further and investigated neural networks to process touch data from all around the phone's surface [38], showing a CNN that identifies fingers upon touch while locating their 3D position.

2.2 Touch accuracy

A sizable effort of HCI research on touch input has gone into modeling input accuracy based on imaged observations. Touch devices generally derive input locations from the center of gravity of recorded intensities. Several studies have found the impact of finger angles on centroid locations and quantified their effect on input accuracy. Benko et al. noticed that centroid-based sampling leads to noticeable error offsets on touch tables [3], and instead proposed deriving input locations from the top of each touch contact. Forlines et al. reported similar observations, tracing the origin of the error back to the flat finger angle participants used when stretching across the surface [17]. This input error is less evident, though still present, on devices that detect pure contact, as Wang and Ren showed on an FTIR-based setup [56]. Following these observations, Holz and Baudisch investigated the impact of finger pose on capactive touchpad recordings, which produced errors that exceed Wang and Ren's by almost 50%, showing the difference between sensing *true* contact and capacitive touch [29, 30]. Finally, Henze et al. collected data from a large population on their tap behavior and modeled error offsets using a data-driven method dependent on screen targets [26]. Their large-scale experiment also nicely showcased the impact of target location on the display on input errors, which is another consequence of varied finger angles.

2.3 Computer-vision based super-resolution

Super-resolution algorithms aim to reconstruct a high-resolution image from one or a set of observations with lower-resolution [43]. These observations typically lack fine-grained details due to hardware and physics limitations in the measurement process [43]. Super-resolution algorithms have found applications in a wide range of fields, from satellite imaging [43] and digital holography [63] to medicine [8], and compressed image enhancement [19].

While several approaches exist that operate on a sequence of low resolution images [15, 16], we focus on the scenario where only a single image is provided as input, a problem known as single image super-resolution (SISR). SISR approaches rely on a strong prior that is learned from a training corpus, adopting the example-based strategy [62]. In recent years, deep learning-based methods

[1]The code and instructions for accessing the complete data corpus can be found at https://siplab.org/projects/CapContact.

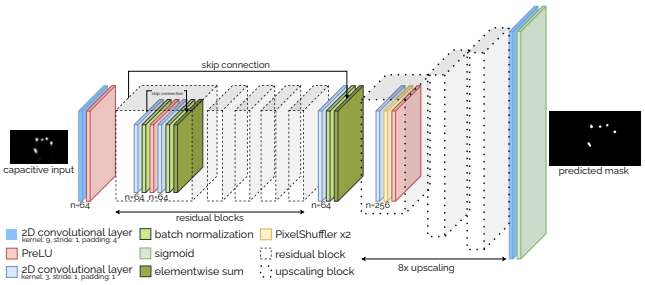

Figure 3: Our generator is a VGG-style fully-convolutional network [39]. Five residual blocks extract the relevant features from the low-resolution capacitive input. Three consecutive sub-pixel convolution layers then upsample it 8×.

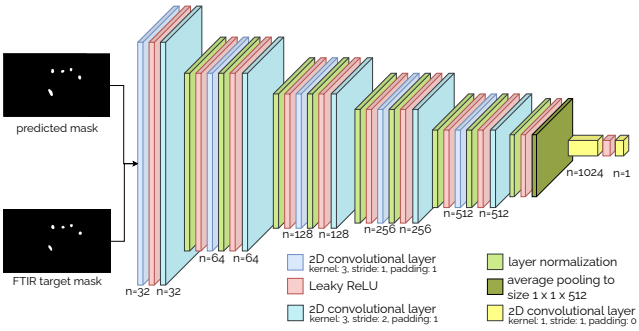

Figure 4: Our critic is based on a VGG-style convolutional neural network [39]. Instead of batch normalization, we apply layer normalization to match the WGAN-GP loss [8]. Our network uses ten convolutional layers to reduce the dimensions of the input image, before it reduces the input to a single dimension using average pooling.

have gathered attention due to their strong performance (see Anwar et al.'s survey on deep learning-based methods for a comprehensive review [1]). Dong et al. were first to train a CNN end-to-end to refine an image that was upsampled using bicubic interpolation [12]. Kim et al. later introduced a VGG-inspired network [34] as well as recursion and skip connections for SISR [35]. The introduction of the super-resolution generative adversarial network (SRGAN) [39] led to another increase in perceived quality and sparked a plenitude of proposed architectural innovations [57, 58, 65].

3 PROPOSED METHOD

We now describe our novel method to predict a high-resolution contact area of a finger touch from a single frame of mutual-capacitance intensities as input. We argue that this problem resembles single image super-resolution as we attempt to reconstruct fine-grained shape information of each finger's contact area from a lower-resolution, quantized two-dimensional capacitive image. Given the success of deep learning-based approaches for single image super-resolution on RGB images [13], we devise a neural network architecture that we refer to as 'generator' to find a valid mapping from the capacitive input space to a corresponding contact shape mask.

3.1 Problem definition

Our method estimates a high-resolution image of the contact area $I^{T_{HR}}$ of a touch from a capacitive sensor's low-resolution frame $I^{C_{LR}}$ as input.

We represent the output of the capacitive touch screen, $I^{C_{LR}}$, as a one-channel real-valued tensor of size $L \times H \times 1$ (similar to Ledig et al. [39]). Our generator G_{θ_G} maps $I^{C_{LR}}$ to $I^{T_{HR}}$, an upscaled mask of the contact area described by a binary tensor of size $rL \times rH \times 1$. Here, r represents the upscaling factor. A pixel value of 1 in the binary contact mask corresponds to a direct contact between the touch surface and the human finger tissue on the corresponding location. A value of 0 implies that no direct contact is established.

Our generator G_{θ_G} consists of a multi-layer feed-forward convolutional neural network, where θ_G represents the weights of the network. Using standard backpropagation, the training procedure adjusts the weights to minimize a loss function l, which quantifies the difference between the predicted contact masks,

$$I^{P_{HR}} = G_{\theta_G}(I^{C_{LR}}), \tag{1}$$

and their corresponding ground-truth contact images $I^{T_{HR}}$, which are available during training.

3.2 Network Architecture

As shown in Figure 3, our generator G_{θ_G} is based on *SRGAN* [39], comprising a CNN with five residual blocks at its core [20]. We chose SRGAN due to its strong performance on SISR datasets with a reasonable number of parameters [1]. The input layer is a 2D convolutional layer with a kernel size of 9, which contributes to a larger receptive field that covers a typical finger touch region in the capacitive image. The residual blocks are connected to three consecutive sub-pixel convolution layers [52] that conduct the upsampling with a factor of 8. Due to its fully-convolutional architecture, we can feed capacitive frames of different sizes into G_{θ_G} as input.

3.3 Loss Function

We train our generator to minimize the objective function

$$l_{MSE-Adv} = l_{MSE} + 0.001 \times l_{Adv}. \tag{2}$$

l_{MSE} is the per-pixel weighted mean squared error (*MSE*) between the target mask $I^{T_{HR}}$ and the predicted contact area $I^{P_{HR}}$,

$$l_{MSE} = \sum_{x=1}^{rL} \sum_{y=1}^{rH} W_{x,y} (I^{P_{HR}}_{x,y} - I^{T_{HR}}_{x,y})^2, \tag{3}$$

where x and y define the coordinates in the high-resolution mask. W is a real-valued tensor of size $rL \times rH \times 1$ carrying the weighting for each pixel difference. Its purpose is to account for the imbalance in the number of contact area pixels (1) and background pixels (0) in the target mask $I^{T_{HR}}$.

We defined and heuristically optimized an algorithm for creating a suitable weighting W that heavily penalizes the false negative prediction of contact pixels. A uniform weighting of the MSE loss term would otherwise lead the network to generate a mask with only 0 pixels due to the few non-zero touch pixels.

First, we define a uniform tensor with values of 1 and then focus on the regions that are outlined by a bounding box enclosing a touch

area in the capacitive image. We add a 2D Gaussian distribution with a standard deviation equal to the lengths of the sides of the bounding box. Its kernel is centered at the centroid of the bounding box and has dimensions of twice the size of its standard deviation. The Gaussian kernel is scaled by factor α times the total area of the covered region. In a final step, we normalize the resulting weight tensor by its total sum. With the smooth Gaussian roll-off from the center of the bounding box we gradually reduce the influence of false predictions in the loss function which facilitates a stable training process.

To encourage G_{θ_G} to create clearer shape outlines, we added an adversarial loss term l_{Adv} to the objective function,

$$l_{Adv} = -D_{\theta_D}(G_{\theta_G}(I^{C_{LR}})). \quad (4)$$

We implement the critic D_{θ_D} as a VGG-style convolutional neural network based on SRGAN Ledig et al. [39]. In contrast to their approach, which implemented a standard GAN, we implement a Wasserstein GAN with Gradient-Penalty (*WGAN-GP*) to increase training stability [23]. Thus, we adapt D_{θ_D} to the requirements of the WGAN-GP [8] and add two further layers to match the size of our high-resolution masks. The critic is trained with the standard WGAN-GP loss defined as

$$l_C = D_{\theta_D}(I^{P_{HR}}) - D_{\theta_D}(I^{T_{HR}}) + \lambda(||\nabla_{\hat{I}} D_{\theta_D}(\hat{I})||_2 - 1)^2, \quad (5)$$

where $\hat{I}$ is uniformly sampled along straight lines connecting $I^{P_{HR}}$ and $I^{T_{HR}}$ [23].

4 DATA CAPTURE

We designed our method for supervised learning and thus require a training corpus with matched pairs of capacitive touchscreen frames and high-resolution images of the corresponding contact area. Since no such dataset currently exists, we devised a data capture apparatus and method to record a suitable set of samples.

Previous research on super-resolved RGB images obtained the low-resolution data by applying bicubic downsampling on the high-resolution images [13, 39]. However, this approach is not applicable to our problem. First, the resolution of a capacitive sensor is typically too low to allow for further degradation. Second, and perhaps more importantly, it is uncertain whether a bicubic kernel correctly captures the relationship between the finger's contact area and the intensity map resolved by the digitizer which is also influenced by the adjacent parts of the finger that hover just above the sensor.

To obtain ground-truth contact areas between each touch and the surface at a high resolution, we based our approach on frustrated total internal reflection, an optical touch-sensing approach that produces a sharp contrast between touching and non-touching parts of the user's finger [25]. Specifically, FTIR enables us to accurately resolve and extract the contact area from each touch, which lights up substantially and is easily detected through simple thresholding.

4.1 Apparatus

We constructed a data capture rig to integrate mutual-capacitance sensing and FTIR into the same touch surface as shown in Figure 5. The key challenge was to build a setup that adds no spacing to the capacitive sensor, so as to ensure that the measurements taken are representative of commodity devices.

To record mutual-capacitance measurements, we mounted an ITO-based transparent touchscreen overlay onto a table-like construction from aluminum profiles at a level of 1.15 m above the floor. The area covered by the ITO diamond gridline pattern measured 15.6″ across, with 345 mm × 195 mm per side. The digitized image had a resolution of 72 px × 41 px, corresponding to the industry-standard pitch between gridlines. Similar to commercial touch devices, the ITO pattern was bonded onto a transparent substrate and covered with a protective sheet of glass on top.

The sensor connected to a digitizer (Microchip ATMXT2954T2), which resolved changes in mutual-capacitance at 16-bit precision. We implemented the communication based on a driver from previous work [32, 53] using Microchip's debug interface. Because the chip returned 16-bit measurements, the update rate topped out at 5 fps due to bandwidth constraints of the communication channel.

In comparison, typical mutual-capacitance digitizers record at 8-bit precision. When a finger approaches, the mutual capacitance drops, which digitizers represent with increased intensities that they clip at 2^7. In contrast, our 16-bit dynamic range does not just increase the resolution of measurements, it also removes the clipping, such that recorded intensities never max out in practice.

Because the protective glass had a thickness of 1.5 mm, we could not simply mount the FTIR sensor on top, lest we sacrifice dynamic range and capacitive sensing quality. Instead, we flipped the capacitive sensor, such that the thin substrate of bonded ITO was exposed to the top. Onto this, we attached a sheet of Plexiglas with 0.75 mm thickness that brought touches to a comparable distance to the ITO on either side. In addition to measuring individual thicknesses, we confirmed that the distance to the ITO sensor added by the Plexiglass produced similar intensities for touch contacts as such touches do on the sensor's protective surface. We also verified the quality of captured intensities across the whole surface. We completed the otherwise traditional FTIR setup with two strips of 15 LEDs (QBLP674-IWM-CW) along the two longer edges.

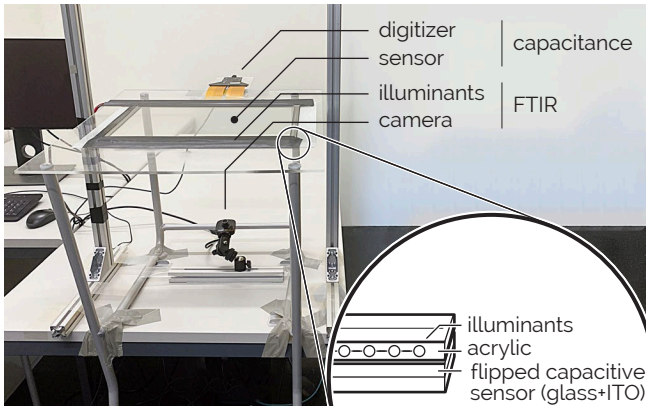

Figure 5: Our apparatus integrates mutual-capacitance touch sensing and optical touch sensing using frustrated total internal reflection to capture accurate contact areas. We flipped the capacitive sensor to expose the ITO layer to the top. A sheet of Plexiglas atop completed the FTIR setup.

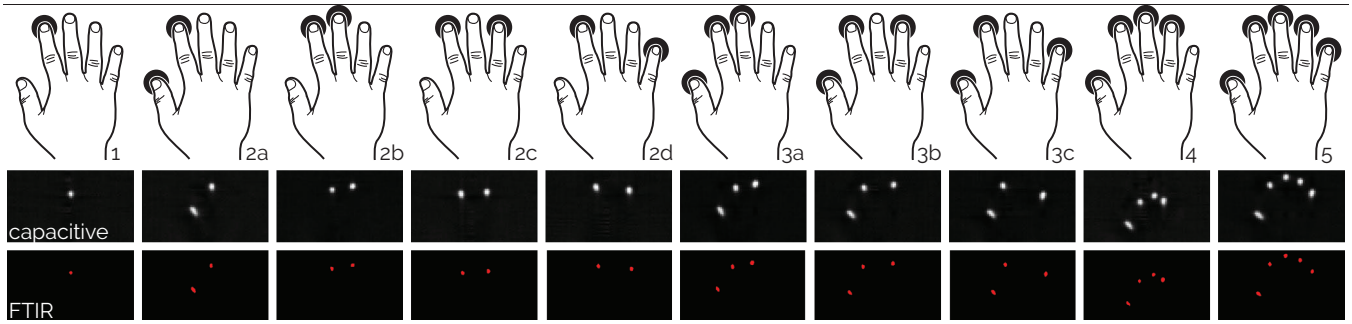

Figure 6: In our data capture study, participants produced a series of touch input and drag events using each of these finger combinations (black circles indicate touches), once at 0° yaw rotation and once at 90° yaw rotation. The figure includes representative capacitive images and their corresponding contact area images from the FTIR sensor for each pose.

To capture contact areas, we mounted an RGB camera 0.4 m below the touch surface (oCam-1CGN-U-T). The camera had a resolution of 1280 px × 960 px at a framerate of 30 fps and we adjusted it to capture the entire area of the touch surface. We calibrated the camera using OpenCV, resulting in the intrinsic and extrinsic parameters to rectify captured images. We then performed four-point calibration to establish a mapping from the touch surface to the camera's image plane. The part of the image that corresponded to the touch surface measured 576 px × 328 px, hence 8× higher than the resolution of the capacitive sensor.

We developed a software application that integrated the drivers and communication with both sensors. Our software featured additional calibration controls, i.e., for the settings of the mutual-capacitive digitizer and camera attributes, and configured the touch digitizer to disable all available on-chip routines, filtering and noise compensation techniques. High-frequency electromagnetic noise filtering remained active and could not be disabled. However, we deactivated detrending, which removes static artifacts from impurities such as rain drops and surface grease, because it would otherwise falsify "raw" readings following resting contacts. The application also temporally synchronized both sensors and ensured continued aligned image capture at a framerate of 5 fps, limited by the SPI throughput of Microchip's ATMXT2954T2 digitizer for offloading capactive images from the chip.

4.2 Participants

For the data collection, we recruited 10 participants from our institution (two female, ages 22–35 years, mean = 28 years). We measured three anatomical characteristics in participants: 1) length of middle finger (75–92 mm, mean=8.3 mm), 2) length of the right hand (176 mm–209 mm, mean=190 mm), and 3) width of the middle finger at the distal joint (16 mm–20 mm, mean=18.5 mm). Each participant received a small gift as a gratuity for their time.

All participants were European and compared to global statistics on hand anatomies, participants' hand lengths came close to the average hand length observed for a similarly aged group of participants from Germany (age: mean=23.8 years, length: mean=183 mm) and China (age: mean=22.9 years, length: mean=183 mm) [45] and covered a wide range of the hand lengths observed in India (men: 169–225 mm, mean=192 mm; women: 143,mm–198 mm, mean=174 mm; [10]). The anatomical characteristics of our participants also closely matched the hand lengths (mean=18.5 mm) as well as the middle finger widths (mean=17.35 mm) and lengths (mean=7.9 mm) of industrial workers in India [7].

4.3 Task and Procedure

In order to obtain a representative dataset comprising a variety of touches, participants performed ten different touch gestures using their right hands using each of the combinations of fingers shown in Figure 6. For each combination of fingers, participants performed a sequence of actions that consisted of placing the corresponding finger tips on the touch screen for about a second, followed by lifting the fingers, placing them back onto the surface, slowly dragging them towards the bottom of the surface, stopping and pausing, and lifting them up. Participants then touched the surface again and slowly dragged their fingers upwards this time.

Before the beginning of the study, the experimenter explained and demonstrated the task to participants. While performing the task, participants only received instructions on which gesture and action to perform next. The experimenter encouraged participants to vary finger pitch angles as well as the touch locations on the surface, but gave no clear instructions about exact angles, positions or pressure. We expected participants to exert similar forces as they would on phones or tablets and that our tasks would naturally lead to variations in the applied pressure, as they included stationary touches as well as dragged poses.

Participants repeated the task with all ten finger combinations for two yaw angles: 0° and 90°. In the 90° condition, their elbow pointed to the right and they dragged their fingers parallel to the long edge of the surface to the right and later to the left, respectively.

Participants repeated the two yaw conditions for all combinations of fingers each for a total of three blocks. At the end of each block, participants performed 20 random touch gestures, comprising both hands simultaneously as well as combinations of fingers and finger angles of their choosing.

Between blocks, participants rested for one minute, letting their hands hang loose. In total, we took recordings of 1200 different gestures, i.e., 10 participants × 3 blocks × (10 gestures × 2 orientations + 20 random gestures). Participants completed all three blocks of the data capture study in under 40 minutes.

4.4 Data filtering

After recording all participants' touch contacts, we processed the data in batch to ensure integrity and quality and to exclude unsuitable images from our training procedure. To obtain accurate masks from touch contacts, we first thresholded the 16-bit signed mutual-capacitance frames and calculated the spatial properties that defined the location (i.e. centroid and bounding box) of each of the resulting connected components. Through scaling contact dimensions, we found the corresponding touch contacts in the FTIR image, from which we extracted a binarized mask of the lit up contact area using Otsu's thresholding [44].

We took several steps to ensure data quality. First, if the number of connected components across both images, capacitive and FTIR, did not match, we discarded the frame. Second, if the contact area of a touch in the FTIR image was larger than the upscaled area derived from the thresholded capacitive image, we also discarded the frame. These filters ensured that only well-detected contact masks were added to our dataset. Each collected data sample consists of the 72 × 41 16-bit output of the capacitive digitizer and a matching binary contact mask with dimensions of 576 px × 328 px.

After processing all recording sessions and discarding frames that are empty (i.e., no touch present) and that are inadmissible following the criteria above, our final dataset comprises over 26,000 pairs of frames. The dataset was thus suitable for our proposed method to serve as input into our machine learning pipeline.

5 EVALUATION

CapContact aims to estimate the contact area of a finger's touch from a capacitive sensor's output. To assess its effectiveness, we compare it against a baseline using bicubic interpolation. To get the baseline image, we upscale the low-resolution capacitive image to the same resolution that we obtain from our FTIR setup and retrieve the contact area using a threshold at 50% of the clip intensity of standard digitizers. In their implementation, this threshold ensures that all touches (i.e. proximal fingers) are detected.

Learning an optimal threshold. Since our collected dataset contains ground-truth contact masks for each capacitive frame, we could also derive an optimized threshold for a selected evaluation metric. Therefore, we introduce a computational resource-saving alternative in the form of a learned threshold that we apply on the upscaled image. We then compare all three methods: bicubic upscaled capacitive image with a naïve threshold, our learned threshold on the upscaled image, and our super-resolution method CapContact.

5.1 Evaluation metrics

As shown in Figure 7, we compare all methods using metrics that quantify the difference between the predicted contact area and the ground-truth mask that stem from FTIR recordings.

Intersection over Union (*IoU*) is a scale-invariant metric commonly applied to measure the similarity between two shapes [49] and defined as

$$IoU(A, B) = \frac{|A \cap B|}{|A \cup B|}, \tag{6}$$

where $A, B \subseteq \mathbb{S} \in \mathbb{R}^n$ are the two shapes to be compared. We obtain IoU by dividing the number of pixel that are correctly classified as area of contact in the predicted mask by the size of the union of the contact areas from the ground-truth and the predicted mask.

The location of the contact area's centroid is important as it defines the touch location. Thus, to understand the quality of our prediction we compare the distance between the predicted centroid and the center of the ground-truth contact area. The centroid can be obtained as the weighted mean of the contact area's pixel coordinates. The ground-truth image and the area predicted by our method are binary masks. Therefore, each pixel receives a uniform weight. To minimize the centroid drift in the thresholded capacitive image due to hovering tissue accounted as contact area, its pixel coordinates are weighted by the capacitive coupling intensity measured at the corresponding location.

Besides the euclidean distance, we analyze the relative position of the predicted centroid relative to the ground-truth centroid. To adjust for yaw, we rotate each error offset—pointing from the target centroid in the FTIR mask to the centroid of the predicted contact mask—by the orientation angle that aligns the major axis of the touch with the y-axis of our coordinate system (Figure 7). We report the coordinates of the resulting vector as Δx and Δy.

Aspect ratio is another important property of a shape: the ratio between the length of its major and minor axis, which is equivalent to the ratio of the two ordered eigenvalues of the weighted covariance matrix. We use the same weighting for the computation of the covariance matrix as for the weighted centroid.

We also analyze the percentage difference between the size of the predicted and the ground-truth area—an IoU-related metric.

We now apply the previous metrics on a *per-touch* basis. To reject noise, we only consider areas in the predicted contact masks as touches that consist of more than 16 connected pixels (in a 2-connected sense). This corresponds to the size of the area covered by a single pixel in the capacitive frame before the upscaling operation. For each predicted touch, we find its nearest-neighbor touch in the matching ground-truth mask based on the distance between their weighted centroids. If two predicted touches share the same ground-truth contact area as their closest reference point, we reject the one with the larger centroid distance as a false-positive prediction (*FP*). We calculated the metrics for and averaged them across the remaining true-positive touches (*TP*) from all samples of the set in relation to their closest neighbor in the target image.

Based on this selection, we consider *precision* and *recall* to assess the reliability of our method. Precision decreases as a method starts to hallucinate touches in the predicted mask that do not have a corresponding touch in the target image—an error that could possibly be introduced by our generative method. Recall indicates the ratio

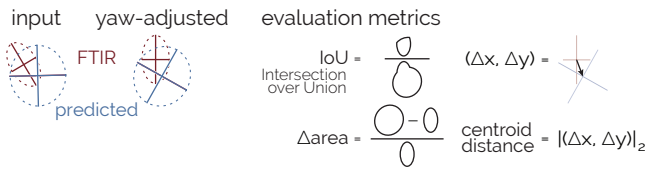

Figure 7: We compare predictions to the ground-truth contact mask using these metrics. For offsets, we first adjust for yaw. Not shown: percentage difference in aspect ratios.

of target touches correctly identified as such. FN is the number of target touches that are not regarded as nearest neighbor by any touch in the predicted image. The F_1 score combines the latter two statistics by computing the harmonic mean,

$$F_1 = 2 \times \frac{\text{recall} \times \text{precision}}{\text{recall} + \text{precision}} \tag{7}$$

5.2 Training details

We implemented CapContact's network using PyTorch and trained it on an *Nvidia GeForce RTX 2080 Ti* with a batch size of 8. During training, we augmented our data online by randomly flipping each sample by either the x, the y, or both axes. We clipped the negative values from the capacitive input frames at 0 and normalized the values of the frame to the range from 0 to 1 by dividing through a constant ceiling value that was larger than all values in the dataset of capacitive samples. We used the Adam optimizer [36] for gradient-based optimization with a learning rate of 5e−5, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. Moreover, we empirically determined $\alpha = 20$ as scaling factor in the creation of the weighting W. We trained for more than 50 training epochs—where each epoch considers the complete training set—and stop the training when IoU with a binary threshold at 0.9 has not increased for more than five epochs on the validation set. We then search the binary cutoff that produces the highest IoU on the validation set with a recall bigger than 95%.

We pretrained the generator for one epoch using the weighted MSE error only. We adjust the weights of the critic five times per generator update.

Learned threshold. To obtain our learned threshold ξ, we solve

$$\underset{\xi \in \mathbb{N}}{\text{argmax}} \frac{1}{n} \sum_{i=1}^{n} IoU((I_i^{Bi_{HR}} > \xi), I_i^{T_{HR}}), \tag{8}$$

using the Nelder-Mead simplex algorithm. $I^{Bi_{HR}}$ is the capacitive image upsampled using bicubic interpolation and is thresholded at ξ into a binary image. We then compute the IoU score between the FTIR target $I^{T_{HR}}$ and the contact area obtained with the threshold ξ across a respective validation set of size n.

For our experiments, we use two different protocols for splitting our data into a training, validation, and test set.

5.3 Experiments 1–2: Contact area & centroid

5.3.1 Experiment 1: Cross-block split. In this experiment, we randomly sample two blocks from each participant. We use 80% of the recorded samples from each of those blocks for training and the other 20% for validation. The data from the third block is for testing. Training CapContact finished after 55 epochs and we obtained the best validation results with a binary cutoff at 0.7. Applied to a bicubic upsampled image, we learned the threshold $\xi = 358$ as optimal, which leads to the highest IoU on the validation set.

Table 1 lists the results of the three methods evaluated on the test set. When trained with adversarial loss, CapContact outperforms both threshold-based methods on the IoU metric. On average, the sizes of the contact areas predicted by CapContact are less than 3% smaller than the contact areas in the FTIR image. This is considerably smaller than the 15.62% difference achieved through the learned threshold and substantially better than the baseline,

Table 1: Exp. 1—Model performances for the cross-block within-participant validation. CapContact outperforms the two threshold-based models in terms of IoU, the offset from the ground-truth centroid, and the percentage difference in area size and aspect ratio.

method	IoU	Δarea [%]	cen.dis. [mm]	Δaspect ratio [%]	Δx [mm]	Δy [mm]	prec.	recall	F_1
baseline	0.19	467.36	1.88	−3.34	**−0.02**	1.26	0.97	**1.00**	0.98
learned threshold	0.62	15.62	1.77	5.00	−0.13	1.18	**1.00**	**1.00**	**1.00**
CapContact	**0.67**	**−2.18**	**1.33**	**2.41**	−0.03	**0.99**	0.96	**1.00**	0.98

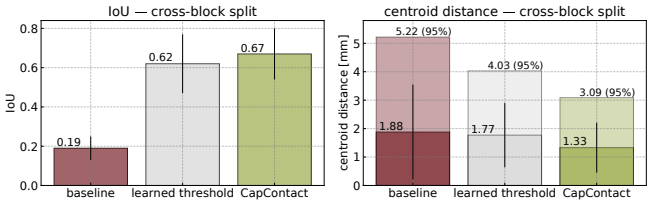

Figure 8: Exp. 1—Tested across blocks within participants, CapContact and the learned threshold achieved a much higher IoU score than the baseline, improving it by factors of 3.5 and 3.2, respectively. On average, CapContact reduced the centroid error by ∼30% and the learned threshold by 6%.

whose contact areas were, on average, almost six times larger than the target areas. The centroids of CapContact's mask predictions are, on average, closer to the ground-truth contact centroids, resulting in the smallest mean offset. On average, its error offsets to ground truth are 0.55 mm smaller than those of the as well as 0.44 mm smaller than those of the learned threshold. All three models achieve a precision, recall, and F_1 score above 95%.

5.3.2 Experiment 2: Cross-person split. In the second experiment, we split our dataset across participants to assess whether our model generalizes to unseen hands. We tested three random folds and averaged the metrics across the three splits:

Fold A included the samples of Participants *2, 3, 6, 7, 8 & 9* for training, *1 & 5* for validation, and *4 & 10* for testing. Fold B contained *3, 4, 6, 7, 9 & 10* in the training set, *2 & 8* in the validation set, and *1 & 5* in the test set. Fold C used *1, 2, 4, 5, 8 & 10* for training, *6 & 7* for validation and *3 & 9* for testing.

Following the same procedure as in Experiment 1, we find CapContact's binary cutoff to be optimal at 0.7 across all three folds. Compared to the cross-block split, we note a slight decrease in IoU for CapContact and a rise in centroid distance for all three methods. Interestingly, the relative error in area size reduced for the two threshold-based methods. The overall trend between the different approaches remained similar to Experiment 1 with CapContact performing the best in terms of IoU and centroid difference. Compared to the baseline, we achieve four times higher IoU, a reduction in centroid distance by 0.41 mm and in Δy by 0.16 mm. The learned threshold method (ξ=358) achieved an IoU score of 0.59. This is worse than the results obtained with CapContact which also reduces the error offset to the ground-truth centroid by 18% compared to the learned threshold.

Table 2: Exp. 2—Model performances for the three cross-participant data-split validations. CapContact predicts contact area sizes and shapes with less than 3% error compared to ground truth. The baseline overestimates the contact area. The learned threshold comes closer, but it yields an error offset that is 0.36 mm larger than CapContact's.

method	IoU	Δarea [%]	cen.dis. [mm]	Δaspect ratio [%]	Δx [mm]	Δy [mm]	prec.	recall	F_1
baseline	0.21	424.06	2.06	−2.90	**0.04**	1.38	0.96	**1.00**	0.98
learned threshold	0.59	6.91	2.01	4.70	−0.15	1.38	**1.00**	0.99	**1.00**
CapContact	**0.63**	**−2.81**	**1.65**	**4.14**	−0.05	**1.22**	0.94	**1.00**	0.97

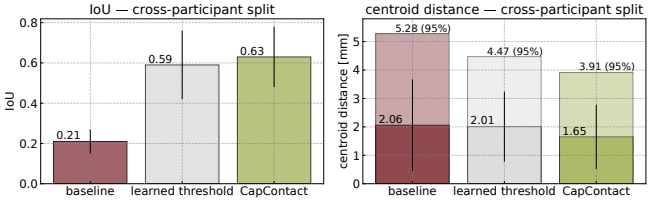

Figure 9: Exp. 2—Tested across participants, CapContact and the learned threshold achieved a much higher IoU score than the baseline, improving it by factors of 3 and 2.8, respectively. CapContact reduced the average centroid error by 20%, whereas the learned threshold reduced it by 2.5%.

Collectively, we have shown that our method has acquired the ability to generalize to unfamiliar hand characteristics, even though we only trained it with a relatively small amount of participants.

5.4 Experiment 3: Two-finger discrimination

We conducted another experiment on our method's performance in discriminating touch contacts that merge in the capacitive image (see Figure 1 right). For validation, we captured new data from four participants: three from the study above (3 male, ages 24–34) and one fresh participant (female, age 24). We recorded the test set with adjacent touches on the same apparatus as described above.

Task and Procedure: During the data capture, participants touched the surface 20 times with both index fingers touching, followed by 20 touches with their thumbs touching. Finally, participants produced another 25 touches on the surface with all combinations of two fingers, one from either hand, touching together. In total, we recorded 570 samples, each of which exhibited two distinct contact areas in the recorded FTIR target image.

Test: For this experiment, we focus on the discrimination of close-by contact areas. We quantify the performance of a method using precision, recall and F_1 score. We hypothesize that our deep learning-based method is capable of extracting individual contact areas from measured intensity distributions, whereas threshold-based approaches fail in discriminating adjacent touches. To process the newly captured data, we used the networks from Experiment 2 for inference, ensuring that the we did not apply a model trained on a fold including a participant in the respective training set.

Table 3: Exp. 3—Full-resolution discrimination of adjacent touches. Two touches were present in each sample. Recall is the quota of fingers detected in the capacitive image, thus a recall of 0.5 means that only one finger was detected an a recall of 1 means that all distinct fingers were extracted.

method	precision	recall	F_1
baseline	0.97	0.53	0.69
learned threshold	1.00	0.72	0.83
CapContact	0.93	**0.93**	**0.93**

Results: Table 3 shows the results of the three methods. Using MSE adversarial loss, our convolutional neural network achieves the highest F_1 score with a recall and precision of 0.93.

Compared to this, the learned threshold achieves a recall of only 0.72, indicating that it failed to individually detect all distinct contact areas for every second frame. The baseline performs the worst. While it achieves a precision of 0.97, it reaches a recall of 0.53. Thus, on average, it detects only one (merged) finger contact in capacitive images, whereas two adjacent were present in each frame. These results imply that a constant threshold is not capable of reliably discriminating individual touches from overlapping regions in the capacitive frame.

5.5 Experiments 4–6: half-resolution sensor

Above, we evaluated our methods using an industry-standard grid-line pitch, which is implemented in our apparatus. Hypothesizing that larger pitches in combination with our method could reach the performance of current systems operating on the standard pitch size, we tested how our approach scales to touch devices with only half of the density of sensor electrodes.

Before testing CapContact's performance on a sensor with that grid-line spacing, we first simulated half-resolution data for training and evaluation purposes by downscaling the recorded capacitive frames and contact masks of our recorded dataset by a factor of 2. By taking the mean across areas of 2×2 pixels and discarding the last row in the original capacitive frame as well as the corresponding rows in the contact mask, we obtain downscaled images of resolution 36 px × 20 px for the capacitive sensor and 288 px × 160 px for the FTIR target.

5.5.1 *Experiment 4: Contact area and centroid at half-resolution.*

We repeated Experiments 2 and 3 on the downscaled dataset. We find the binary cutoff for CapContact to be optimal at 0.5 which achieves the highest IoU on the validation set. The learned threshold amounted to $\xi = 256$ to maximize the same metric.

Table 4 reports the results of the three methods averaged across the three cross-participant splits from Experiment 2. Our network CapContact infers the contact area with an IoU of 0.63 and produces an error offset of 1.73 mm, matching the performance on the standard resolution screen in Table 2. The efficacy of the threshold-based methods deteriorates on the lower-resolution capacitive frames. The baseline accomplishes an IoU of only 0.17—performing almost four times worse than CapContact. The baseline's error offset amounts to 2.21 mm. The learned threshold predicts the contact masks whose

Table 4: Exp. 4—*Half*-resolution model performances: We follow the cross-participant protocol from Experiment 2. CapContact achieves comparable performance on the downscaled dataset while the IoU score deteriorates for both threshold-based methods.

method	IoU	Δarea [%]	cen.dis. [mm]	Δaspect ratio [%]	Δx [mm]	Δy [mm]	prec.	recall	F_1
baseline	0.17	554.58	2.21	−7.62	−0.25	1.55	0.99	0.99	**0.99**
learned threshold	0.53	30.71	2.27	**−0.67**	−0.19	1.49	**1.00**	0.95	0.98
CapContact	**0.63**	**1.97**	**1.73**	1.29	**−0.09**	**1.25**	0.99	**1.00**	**0.99**

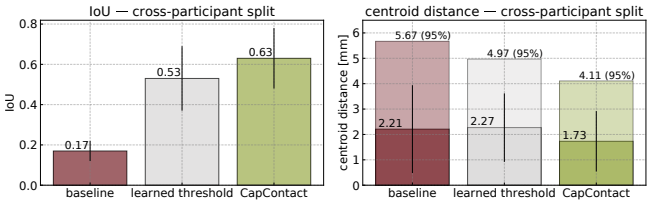

Figure 10: Exp. 4—*Half*-resolution performance on touch metrics: CapContact and the learned threshold achieved an average IoU score that is 3.7× and 3.1× higher than the baseline, respectively. CapContact lowered the average centroid error by 22%, while the learned threshold was 3% worse.

error offset is 30% worse than CapContact's. It also achieves an IoU of 0.53 (15.9% worse than CapContact) and an average area that is 31% too large compared to the actual contact.

5.5.2 Experiment 5: Discriminating closely adjacent fingers at half-resolution. As shown in Table 5, our proposed method still discriminates close-by contact areas with a precision of 0.95, a recall of 0.78, and an F_1 score of 0.85. Compared to this, both threshold-based methods fail to distinguish nearby fingers on the downscaled dataset from Experiment 3, achieving an F_1 score of 0.69 and a recall of only 0.53. The discrimination ability of CapContact on the lower-resolution sensor even outperforms the two threshold-based methods applied on the data with standard resolution.

5.5.3 Experiment 6: Testing CapContact's generalizability on an unseen sensor with a larger pitch. To assess if our conclusions from Experiment 4 transfer to real-world settings, we investigated CapContact's capablility of inferring contact areas from the capacitive images captured on a larger-pitch sensor. For this purpose, we used a Project Zanzibar mat [53]. The flexible sensor surface has a dimension of 42 × 30 cm with a sensor resolution of 58 × 41 pixels, resulting in a pitch of 7 mm—almost twice as large as the industry standard implemented in touch-screen devices. *Without fine-tuning* on new data, we reused the network we trained on the downscaled training set of Fold A in Experiment 4 for inference to conduct this experiment. Because of Project Zanzibar's opaque sensor, we could not simultaneously record ground-truth FTIR images and, thus, resorted to a qualitative analysis of the predicted contact masks.

Figure 11 shows the capacitive images acquired from Project Zanzibar's sensor surface for four example cases. Even though the training data only contained simulated low-resolution data and no actual samples from the Zanzibar mat, CapContact reliably detects

Table 5: Exp. 5—*Half*-resolution discrimination of adjacent touches on downscaled frames from Exp. 3 with two touches in each frame. A recall of 0.5 means that only one finger was detected, a recall of 1 means that all fingers were detected.

method	precision	recall	F_1
baseline	1.00	0.53	0.69
learned threshold	1.00	0.53	0.69
CapContact	0.95	**0.78**	**0.85**

individual touch inputs and discriminates contact areas that appear to overlap in the capacitive frame (Figure 11b&c). More astonishingly, the network robustly detects contact areas even under noisy conditions due to hover and small deformations in the malleable surface as occurred for four fingers (d), successfully rejecting noisy inputs and solely preserving actual touch contacts.

CapContact's performance is in stark contrast to the baseline, which overestimates the contact area (a), merges contact areas in the produced outputs (c) & (d), and confuses noise with touch input (d), indicating that the threshold is too low.

While the learned threshold filters out noise (d) and produces a more reasonable contact area (a) due to its higher cutoff, it simultaneously leads to missed events from smaller fingers (e.g., left pinky (d) and softer touches (b)), hinting that the threshold is too high. However, the same threshold causes the contact areas of the adjacent finger contacts to fuse (c&d), indicating that for these

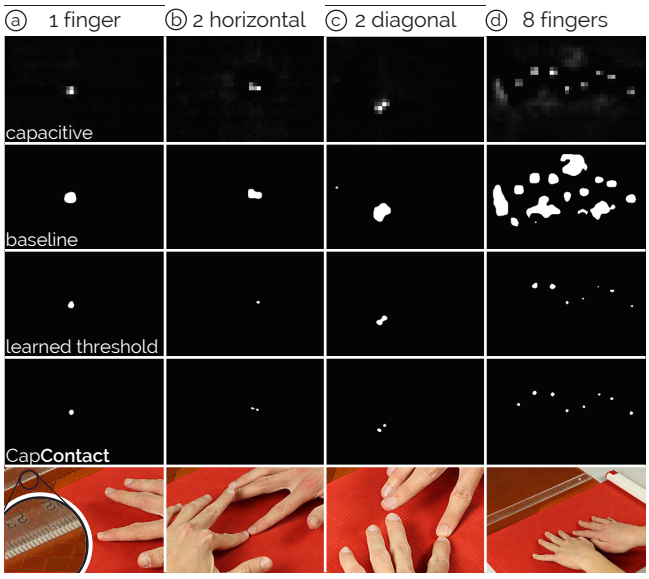

Figure 11: Exp. 6—CapContact running at half resolution (Project Zanzibar mat [53], 7 mm pitch). Solely trained on down-sized data from our capture study, CapContact robustly detects individual contact areas while all threshold-based methods suffer from noise and missed touch events.

cases, it is still too low. This contradiction arises from the complex relationship between contact area, finger shape, and capacitive coupling which cannot be reconciled by just a constant threshold.

Although we investigated data from this experiment empirically, observing CapContact's output at such a level of quality provides support that our method can generalize.

6 DISCUSSION

Our findings demonstrate that CapContact achieves superior performance in predicting high-resolution contact areas from low-resolution capacitive images compared to all threshold-based estimations. Trained with an MSE adversarial loss, CapContact predicts the size of the contact area with less than three percent relative error on average. It also reduces the error offset between the weighted centroid of the capacitive image and the true center of the contact area by more than 20%. As detailed above, CapContact manages to reliably separate touch areas that blend together in the capacitive input—arguably the core task to perform next to detecting touch events and locations. This ability of our SRGAN-adapted generator is likely acquired through the adversarial training that encourages the generator to create contact areas with known shapes. This observation justifies the validity of our adversarial-based loss function.

As capacitive measurements are not limited by a practical maximum intensity in the case of our 16-bit digitizer and rarely span the whole range, when using a threshold it must ensure that all touches are detected. For example, a palm results in higher values than single fingers due to the non-linear coupling to the larger skin surface (e.g., as shown in Guarneri et al.'s Figures 8–9 [22]). This results in a trade-off threshold-based methods must make between reliably detecting (small) touches, reliably distinguishing adjacent touches, and reliably rejecting erroneous input in the form of noise.

Through the dataset we collected, we could determine an optimal threshold with regard to approximating actual contact areas. While this learned threshold achieved a higher IoU than the baseline threshold in Experiments 1 and 2, it still performed worse than CapContact and failed to reliably discriminate adjacent fingers in Experiment 3. On top of that, using a value as high as $\xi = 358$ as a hard-coded, shape-agnostic threshold risks missing touches of small fingers with tiny contact areas (e.g., by kids). That is, much like the baseline threshold, the learned threshold is also subject to the trade-off we described above.

In comparison, we saw in our evaluation that CapContact is little prone to facing this trade-off. Especially for two-finger separation of closely adjacent touches, CapContact achieved a reliable recall (0.93) compared to the threshold-based methods that were far below (0.53 and 0.72). Relating this to the successful cases of distinguishing touches, this leaves the baseline with only 43 of 570 successful separations (7.5%). The learned threshold fared better at 255 of 570 (44.7%), but still far below CapContact's 494 of 570 (86.7%).

Finally—and most surprisingly—CapContact reliably operated on half the sensor resolution across the same surface area. Mean IoU rates remained at 0.63, whereas those of the baseline and learned threshold dropped by 20% and 4%, respectively. The starkest contrast perhaps was in CapContact's capability of still separating closely adjacent touches. Here, we found a success rate of 314 of 570 (55.1%), which was superior to the baseline's 32 of 570 (5.6%) and the learned threshold's 36 of 570 (6.3%). The threshold-based methods are thus inadequate for reliable detection and discrimination of fingers as well as rejecting noise at this higher grid-line spacing justifying the use of a stronger function approximator such as our neural network CapContact.

In our last experiment, we additionally demonstrated that CapContact generalized to unseen recordings from an unseen sensor with a lower sensor density on a larger surface, amounting to almost twice the grid-line pitch that is common in the industry today. Still, in our empirical evaluations and through manual inspection, we observed CapContact's performance in producing reasonable contact areas given the respective capacitive input frame as well as its capability of rejecting noisy inputs due to sensor deformations following touch input, and, importantly, its reliability in differentiating two close-by touches. We also observed that the threshold-based methods lost those capabilities when operating at the larger pitch, as the consequences of the trade-off between noise rejection and touch detection become more challenging at a lower resolution.

For potential future improvements, the absolute IoU scores leave room for improvement in further iterations of our method. While our results show promise for predicting accurate contact-area sizes and shapes, we expect that the key to the enhancement of IoU scores lies in a further reduction in centroid differences. One option in this regard could be formulating the estimation of contact masks as a segmentation problem common in image processing [33].

6.1 Limitations

Our method comes with a couple of limitations. First, our deep learning-based approach requires substantial computational power for training. A single training iteration with a batch size of 8 takes around 1.37 s on average, causing a single epoch on the complete training dataset to take more than one hour. Once trained, the inference time needed for a single batch is only around 2 ms when computed on a GPU. In total, our generator has 861,194 parameters and in future work, we plan to explore model-size reduction.

Next, our collection study captured participants' fingertips at various pitch and yaw poses. We expect that our method extends to varying roll postures, but also surmise that reconstructing more complex shapes accurately may be challenging due to the lack of training data in our corpus. As shown in Figure 12, we recorded a frame where a user holds a stylus while resting the side of the palm on the surface. For the side of the palm, we see two distinct contact areas with irregular outlines in the FTIR image. The two areas blend together in the capacitive image and no threshold can separate the bicubic upscaled blob. Our method, too, merges both blobs, but reconstructs parts of the irregularly shaped contact outline.

The predicted touch mask also generated an additional blob at the bottom left of the palm. While we caught and reported such superfluously generated blobs in our analysis above, the presence of spurious contact data shows that the network tends to hallucinate shapes for unseen capacitive input during training.

6.2 Implications for capacitive touch sensing

Based on our results, we can draw more generalized implications on touch sensing. First and foremost, our method can super-resolve

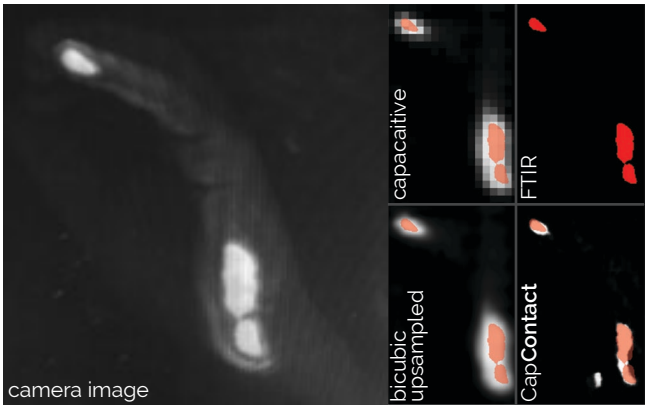

Figure 12: Sample image of hand side resting on the surface when writing with a pen. Since our dataset to-date contains combinations of finger touches at various angles, but does not include more general and complex shapes, the fidelity of shape approximation is limited. While CapContact reconstructs shapes that are more accurate than the bicubic baseline, which produces an expansive imprint, our method still misses the fine crevasse between parts of the palm.

contact masks at eight times the resolution of the standard capacitive sensor with suitable accuracy. This opens the opportunity for lower capacitive sensor resolutions that, in conjunction with our method, could perform the main tasks of today's capacitive sensors, though at reduced sensor resolution and thus more inexpensively. The results of Experiments 4–6 support this conclusion and we see numerous opportunities for host-side processing in future touch devices as well as on embedded systems that use capacitive sensing on passive objects or skin, where few sensing lines exist yet reliable and precise multitouch is desirable.

Second, our method reliably discriminates between adjacent touches whose capacitive coupling intensities merge in the capacitive frame. This capability has previously only been demonstrated using temporal tracking and monitoring contact 'sizes' of blobs in capacitive images [2]. Our method, in contrast, accomplishes this on a *single frame* and, as validated in Experiments 4–6, performs with comparable reliability at lower capacitive sensor resolutions. Given our results so far, we expect that our method's reliability would only increase by adding temporal tracking in the future.

Third, we see a main implication of CapContact in bridging the gap between capacitive sensing—a sensing modality that inherently detects not contact but the *proximity* of fingers and objects depending on their coupling attributes—and the rich body of related work on touch *shapes* and their use. By integrating CapContact in a preprocessing step, such techniques now have the potential to migrate to the capacitive touchscreens on tablets, tables, and other large-screen displays (e.g., [6, 46, 47, 61]).

Our method also has implications for special-purpose input techniques using capacitive sensing as we detail now.

6.2.1 Distinguishing hover vs. contact. CapContact opens up the opportunity for better hover, touch, and pressure distinction, which will allow future work to enrich touch sensing through treating each one of these modalities as a continuous input modality as postulated by Hinckley and Sinclair in the late 1990s [28], yet bringing this to current commodity devices. For finger touches, our method is capable of finely distinguishing between actual contact and parts of the finger that are just above the surface. This fine separation can feed into pipelines that determine finger angles [42, 50] to improve predictions. Trained on a more exhaustive set of touch shapes and additional parts of the user's hand, such as palm, wrist, knuckles, fist, our method could also benefit inadvertent touch rejection when using styli or other tangible objects [64] in the future.

6.2.2 Touch pressure. CapContact also has implications for detecting touch pressure. When applying increased pressure, the finger's outline remains similar, yet the contact area will increase, as accurate contact areas in conjunction with finger widths have proved as a reliable predictor for pressure (e.g., Liu and Yu's comparison of contact area and touch force on a membrane under varying poses [41]). With our method, we obtain the traditional measurements from the capacitive sensor, including the small amounts of hover caused by the capacitive coupling of finger outlines above the surface, in addition to the contact area. These metrics could serve as input into a simple physically-inspired mapping to touch pressure without a holistic calibration procedure.

6.2.3 Touch-input accuracy. As shown in Figure 2 and quantified in our evaluation, CapContact yields touch centroids that are much closer to the actual center of contact. This, in turn, directly impacts the predictive power of our method for accurate touch locations.

Previous work has shown that the impact of finger angles on touch-input accuracy is significant and that, as a consequence, inferring input locations from contact masks [56] leads to a smaller error than when deriving them from the capacitive sensor-based center of gravity [29]. In this context, the results of our experiments quantify the difference between actual contact centroids and center-of-mass locations of capacitive imprints.

Comparing previous findings in terms of minimum button size for reliable activation (i.e., in 95% of cases), Wang et al. established 10.8 mm per side on a contact-based sensor (index finger condition), compared to 15 mm on a capacitive touchpad [29]. Since CapContact recovers actual contact areas from capacitive imprints, our method has engineering benefits for the design of reliable touch user interfaces. As shown in Figure 8 and accounting for 95% of all cases, the discrepancy between the capacitive centroid (baseline) and the ground-truth contact mask is 5.2 mm. CapContact reduces this error by 26% to 3.9 mm independent of participant, while per-user calibration reduces the error by over 40% to 3.1 mm (Figure 9). CapContact maintains its benefits even in our half-resolution experiments and reduces the capacitive baseline's error offset by 28% to 4.1 mm without per-user calibration (Figure 10).

7 CONCLUSIONS

In this paper, we have introduced the first investigation of deriving actual contact shapes from mutual-capacitance sensing—the touch sensing modality that exists on virtually all touch-screen devices. While capacitive sensing was never intended to resolve contact shapes, we have demonstrated its feasibility through our method CapContact and have quantified the precision of reconstructing the

contact shapes between the user's finger and the surface from a single capacitive image using a data-driven approach.

Our contribution comprises a data corpus of 10 participants with paired and registered capacitive images and ground-truth contact shapes that we captured from optical touch sensing based on frustrated total internal reflection. We used the collected corpus to train an 8× super-resolution generative adversarial network for the purpose of refining touch masks from lower resolution capacitive images as input and quantify its performance through a series of experiments. Importantly, not only does our deep learning-based method CapContact reconstruct high-resolution contact masks with less than 3% error in contact area, it can also separate closely adjacent touch contacts that are indistinguishable in the raw capacitive images and, thus, merge into a single touch on existing systems.

We have also demonstrated the potential of our method to evolve mutual-capacitive sensing for touch input as we know it today. Following the results of our evaluations, we have shown CapContact's capability of performing with near-comparable performance on just *half the sensor resolution*. The potential implications of these findings include future touch devices with evolved capacitive sensors, operating at a larger grid-line pitch, thus requiring less sensor material, obtaining higher signal-to-noise measurements, and reducing sensing combinations—all while achieving better contact estimation, noise rejection, and two-point separation than on *today's* full-resolution sensors across the same surface area.

By releasing our method, trained models, as well as our collected data corpus to the community, we believe that our method is a step into future explorations of touch *contact* and *shape* on capacitive touchscreens and that it will lead to advances in natural user interaction as well as touch-input accuracy.

REFERENCES

[1] Saeed Anwar, Salman Khan, and Nick Barnes. 2019. A deep journey into super-resolution: A survey. *arXiv preprint arXiv:1904.07523* (2019).

[2] Hrvoje Benko and Andrew Wilson. 2015. Resolving merged touch contacts. US Patent 9,122,341.

[3] Hrvoje Benko, Andrew D Wilson, and Patrick Baudisch. 2006. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. 1263–1272.

[4] Sebastian Boring, David Ledo, Xiang'Anthony' Chen, Nicolai Marquardt, Anthony Tang, and Saul Greenberg. 2012. The fat thumb: using the thumb's contact size for single-handed mobile interaction. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services*. 39–48.

[5] Alan Bränzel, Christian Holz, Daniel Hoffmann, Dominik Schmidt, Marius Knaust, Patrick Lühne, René Meusel, Stephan Richter, and Patrick Baudisch. 2013. GravitySpace: tracking users and their poses in a smart room using a pressure-sensing floor. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 725–734.

[6] Xiang Cao, Andrew D Wilson, Ravin Balakrishnan, Ken Hinckley, and Scott E Hudson. 2008. ShapeTouch: Leveraging contact shape on interactive surfaces. In *2008 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems*. IEEE, 129–136.

[7] Arunesh Chandra, Pankaj Chandna, and Surinder Deswal. 2011. Analysis of hand anthropometric dimensions of male industrial workers of Haryana state. *International Journal of Engineering (IJE)* 5, 3 (2011), 242–256.

[8] Yuhua Chen, Feng Shi, Anthony G Christodoulou, Yibin Xie, Zhengwei Zhou, and Debiao Li. 2018. Efficient and accurate MRI super-resolution using a generative adversarial network and 3D multi-level densely connected network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 91–99.

[9] Philip L Davidson and Jefferson Y Han. 2006. Synthesis and control on large scale multi-touch sensing displays. In *Proceedings of the 2006 conference on New interfaces for musical expression*. 216–219.

[10] Sangeeta Dey and AK Kapoor. 2015. Hand length and hand breadth: a study of correlation statistics among human population. *Int J Sci Res* 4, 4 (2015), 148–150.

[11] Paul Dietz and Darren Leigh. 2001. DiamondTouch: a multi-user touch technology. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*. 219–226.

[12] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2014. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*. Springer, 184–199.

[13] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2015. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence* 38, 2 (2015), 295–307.

[14] John Elias and Wayne Westerman. 1998. FingerWorks. https://en.wikipedia.org/wiki/FingerWorks [Online; accessed 16-September-2020].

[15] Georgios D Evangelidis and Emmanouil Z Psarakis. 2008. Parametric image alignment using enhanced correlation coefficient maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 10 (2008), 1858–1865.

[16] Sina Farsiu, M Dirk Robinson, Michael Elad, and Peyman Milanfar. 2004. Fast and robust multiframe super resolution. *IEEE transactions on image processing* 13, 10 (2004), 1327–1344.

[17] Clifton Forlines, Daniel Wigdor, Chia Shen, and Ravin Balakrishnan. 2007. Direct-touch vs. mouse input for tabletop displays. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 647–656.

[18] Kentaro Fukuchi and Jun Rekimoto. 2002. Interaction techniques for smartskin. In *Proceedings of UIST*, Vol. 2.

[19] G. Gilboa, N. Sochen, and Y. Y. Zeevi. 2002. Forward-and-backward diffusion processes for adaptive image enhancement and denoising. *IEEE Transactions on Image Processing* 11, 7 (2002), 689–703.

[20] Sam Gross and Michael Wilber. 2016. Training and investigating Residual Nets. (Feb 2016). http://torch.ch/blog/2016/02/04/resnets.html

[21] Tobias Grosse-Puppendahl, Christian Holz, Gabe Cohn, Raphael Wimmer, Oskar Bechtold, Steve Hodges, Matthew S Reynolds, and Joshua R Smith. 2017. Finding common ground: A survey of capacitive sensing in human-computer interaction. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 3293–3315.

[22] I Guarneri, Alessandro Capra, Giovanni Maria Farinella, F Cristaldi, and Sebastiano Battiato. 2014. Multi Touch Shape Recognition for Projected Capacitive Touch Screen.. In *VISAPP (3)*. 111–117.

[23] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. In *Advances in neural information processing systems*. 5767–5777.

[24] Anhong Guo, Robert Xiao, and Chris Harrison. 2015. Capauth: Identifying and differentiating user handprints on commodity capacitive touchscreens. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces*. 59–62.

[25] Jefferson Y Han. 2005. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th annual ACM symposium on User interface software and technology*. 115–118.

[26] Niels Henze, Enrico Rukzio, and Susanne Boll. 2011. 100,000,000 taps: analysis and improvement of touch performance in the large. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. 133–142.

[27] Ken Hinckley, Seongkook Heo, Michel Pahud, Christian Holz, Hrvoje Benko, Abigail Sellen, Richard Banks, Kenton O'Hara, Gavin Smyth, and William Buxton. 2016. Pre-touch sensing for mobile interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 2869–2881.

[28] Ken Hinckley and Mike Sinclair. 1999. Touch-sensing input devices. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 223–230.

[29] Christian Holz and Patrick Baudisch. 2010. The generalized perceived input point model and how to double touch accuracy by extracting fingerprints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 581–590.

[30] Christian Holz and Patrick Baudisch. 2011. Understanding touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2501–2510.

[31] 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*. 3011–3014.

[32] Christian Holz and Marius Knaust. 2015. Biometric touch sensing: Seamlessly augmenting each touch with continuous authentication. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 303–312.

[33] Shruti Jadon. 2020. A survey of loss functions for semantic segmentation. In *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. IEEE, 1–7.

[34] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1646–1654.

[35] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1637–1645.

[36] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[37] Huy Viet Le, Thomas Kosch, Patrick Bader, Sven Mayer, and Niels Henze. 2018. mayer2017estimating. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[38] 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*. 779–792.

[39] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4681–4690.

[40] Seonkyoo Lee. 1984. A fast multiple-touch-sensitive input device. *A Thesis Submitted in Conformity with the Requirements for the Degree of Master of Applied Science in the Department of Electrical Engineering, University of Toronto* (1984).

[41] Na Liu and Ruifeng Yu. 2018. Investigation of force, contact area and dwell time in finger-tapping tasks on membrane touch interface. *Ergonomics* 61, 11 (2018), 1519–1529.

[42] 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*. 220–229.

[43] Kamal Nasrollahi and Thomas B Moeslund. 2014. Super-resolution: a comprehensive survey. *Machine vision and applications* 25, 6 (2014), 1423–1468.

[44] Nobuyuki Otsu. 1979. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics* 9, 1 (1979), 62–66.

[45] Pei-Luen Patrick Rau, Yubo Zhang, Louis Biaggi, Roberto A Engels, Long Qian, and Henrik Ribjerg. 2015. How Large is Your Phone? A Cross-cultural Study of Smartphone Comfort Perception and Preference between Germans and Chinese. *Procedia Manufacturing* 3 (2015), 2149–2154.

[46] Jason L Reisman, Philip L Davidson, and Jefferson Y Han. 2009. A screen-space formulation for 2D and 3D direct manipulation. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*. 69–78.

[47] Jun Rekimoto. 2002. SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 113–120.

[48] Jun Rekimoto, Takaaki Ishizawa, Carsten Schwesig, and Haruo Oba. 2003. PreSense: interaction techniques for finger sensing input devices. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*. 203–212.

[49] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 658–666.

[50] Simon Rogers, John Williamson, Craig Stewart, and Roderick Murray-Smith. 2011. AnglePose: robust, precise capacitive touch tracking via 3d orientation estimation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2575–2584.

[51] Adam Schwartz. 2014. Capacitive Sensing: Looking under the hood. Synaptics Incorporated. http://www.krishnamoorthy.com/comsoc/docs/20140312-Synaptics-Schwartz.pdf

[52] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1874–1883.

[53] 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*. 1–13.

[54] Geoff Walker. 2014. Fundamentals of Projected-Capacitive Touch Technology. Talk at SID Display Week. http://www.walkermobile.com/SID_2014_Short_Course_S1.pdf

[55] Feng Wang, Xiang Cao, Xiangshi Ren, and Pourang Irani. 2009. Detecting and leveraging finger orientation for interaction with direct-touch surfaces. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*. 23–32.

[56] Feng Wang and Xiangshi Ren. 2009. Empirical evaluation for finger input properties in multi-touch interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1063–1072.

[57] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. 2018. Recovering Realistic Texture in Image Super-resolution by Deep Spatial Feature Transform. *CoRR* abs/1804.02815 (2018). arXiv:1804.02815 http://arxiv.org/abs/1804.02815

[58] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. 2018. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 0–0.

[59] Martin Weigel, Tong Lu, Gilles Bailly, Antti Oulasvirta, Carmel Majidi, and Jürgen Steimle. 2015. Iskin: flexible, stretchable and visually customizable on-body touch sensors for mobile computing. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 2991–3000.

[60] Wayne Carl Westerman and John G Elias. 2010. Capacitive sensing arrangement. US Patent 7,764,274.

[61] Andrew D Wilson, Shahram Izadi, Otmar Hilliges, Armando Garcia-Mendoza, and David Kirk. 2008. Bringing physics to the surface. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*. 67–76.

[62] Chih-Yuan Yang, Chao Ma, and Ming-Hsuan Yang. 2014. Single-Image Super-Resolution: A Benchmark. In *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 372–386.

[63] Shuqun Zhang. 2006. Application of Super-Resolution Image Reconstruction to Digital Holography. *EURASIP Journal on Applied Signal Processing* 2006 (01 2006), 238–238. https://doi.org/10.1155/ASP/2006/90358

[64] Yang Zhang, Michel Pahud, Christian Holz, Haijun Xia, Gierad Laput, Michael McGuffin, Xiao Tu, Andrew Mittereder, Fei Su, William Buxton, et al. 2019. Sensing posture-aware pen+ touch interaction on tablets. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–14.

[65] Yuxin Zhang, Zuquan Zheng, and Roland Hu. 2020. Super Resolution Using Segmentation-Prior Self-Attention Generative Adversarial Network. *arXiv preprint arXiv:2003.03489* (2020).