Deep Learning Super-Resolution Network Facilitating Fiducial Tangibles on Capacitive Touchscreens

Marius Rusu
LMU Munich
Munich, Germany
rusu.marius97@gmail.com

Sven Mayer
LMU Munich
Munich, Germany
Munich Center for Machine Learning (MCML)
Munich, Germany
info@sven-mayer.com

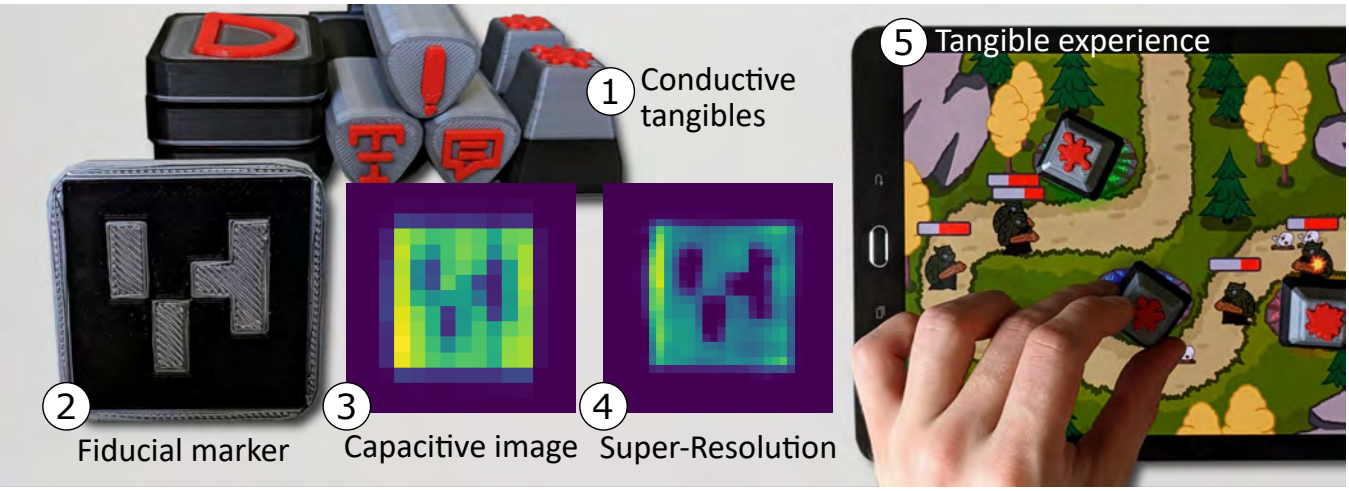


Figure 1: We propose a Super-Resolution Generative Adversarial Network to facilitate tangibles on capacitive touchscreens (5). By equipping conductive tangibles (1) with fiducial markers (2), we gather low-resolution capacitive images (3). The network super-resolves these images (4) and enables off-the-shelf fiducial detection algorithms to track the fiducial markers.

ABSTRACT

Over the last few years, we have seen many approaches using tangibles to address the limited expressiveness of touchscreens. Mainstream tangible detection uses fiducial markers embedded in the tangibles. However, the coarse sensor size of capacitive touchscreens makes tangibles bulky, limiting their usefulness. We propose a novel deep-learning super-resolution network to facilitate fiducial tangibles on capacitive touchscreens better. In detail, our network super-resolves the markers enabling off-the-shelf detection algorithms to track tangibles reliably. Our network generalizes to unseen marker sets, such as AprilTag, ArUco, and ARToolKit. Therefore, we are not limited to a fixed number of distinguishable objects and do not require data collection and network training for new fiducial markers. With extensive evaluation, including real-world users and five showcases, we demonstrate the applicability of our open-source approach on commodity mobile devices and further highlight the potential of tangibles on capacitive touchscreens.

CCS CONCEPTS

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

KEYWORDS

human-computer interaction, deep learning, super resolution, capacitive touchscreen

ACM Reference Format:
Marius Rusu and Sven Mayer. 2023. Deep Learning Super-Resolution Network Facilitating Fiducial Tangibles on Capacitive Touchscreens. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3544548.3580987

CHI '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3580987

1 INTRODUCTION

Touch is the primary means of interaction with a comprehensive set of devices, such as smartphones, tablets, smart appliances [37],

and cars [61]. Nonetheless, touch interaction lacks input expressiveness [70, 75]. Researchers proposed using tangibles to enhance the interactive experience, cf. Grosse-Puppendahl et al. [28]. In detail, researchers explored stackable [3, 7, 31], deformable [71, 84], and touch-sensitive [24, 26] tangibles enabling a wide range of applications, such as architecture [80] and learning [51]. Despite their apparent advantages, commercially available tangible interfaces, such as TangiPlay[1] and Microsoft's PixelSense[2], still remain scarce. On the other hand, today's capacitive touchscreens enable easy and accurate finger tracking [68] using its coarse sensor size (e.g., ~4mm [56]), for instance, via simple centroid estimation [44]. In contrast, it is hard to facilitate tangible tracking using these sensors as detecting many bits is required. Moreover, detection algorithms for fiducial markers are optimized for higher resolutions using RGB cameras. Thus, tangibles on capacitive touchscreens must either be bulky or limit the number of distinguishable objects to allow recognition using the coarse touch sensor. This drastically limits the applicability of tangibles on today's touch devices, despite the many promising applications.

Researchers applied state-of-the-art methods to the raw data of the capacitive sensor allowing tangible tracking on today's capacitive touchscreens. For instance, they used geometric multi-frame super-resolution techniques [56] and super-resolution deep learning [75, 76]. They showed that super-resolution techniques effectively improve the resolution of everyday objects and adjacent touch points. However, today's approaches mostly do not restore the imprint of the tangibles on the screen but directly predict the properties of the objects, e.g., marker id and rotation [70]. Therefore, they neglect the high-quality domain-specific detection algorithms developed in the last decade, such as ArUco [23] and AprilTag [62, 85] detectors. Thus, having a generalizable super-resolution model to super-resolve capacitive fiducial marker imprints would allow us to bring back off-the-shelf detection algorithms to today's capacitive touchscreens.

In this paper, we extend over prior work by proposing a super-resolution generative adversarial network to super-resolve fiducials and facilitate tangibles on capacitive touchscreens using off-the-shelf detection algorithms. Our approach is visualized in Figure 1 and super-resolves single 30×30 pixel capacitive images of fiducial markers to 60×60 pixel. This enables off-the-shelf detection algorithms to track tangibles as small as 24×24mm with a high accuracy of 91.9% and a small rotation MAE of 3.85°. We conditioned the network only on AprilTag 36h11 [62, 85] markers.

Our results show that the network generalizes well beyond these known markers and allows the detection of AprilTag 16h5, ArUco original [23], and ARToolKit 4×4 [38, 83] markers. Addtionally, we show that our approach outperforms a traditional interpolation algorithm (Lanczos-4 interpolation [45]) and a Single-Image Super-Resolution (ESRGAN [86]) baseline. Therefore, our approach is not limited to a fixed number of distinguishable objects and does not require data collection and network training for new fiducial markers. Furthermore, we enable off-the-shelf fiducial detection algorithms to operate on capacitive images and reduce the development effort for custom algorithms. We deploy the network on a commodity tablet for real-time fiducial tracking with 124ms inference duration and perform a real-world user evaluation for tangible interaction showing average detection times between 832ms and 2231ms. We showcase applications for learning, text editing, and gaming, where tangibles improve the interactive experience. We share the model, data, and code in our open-source repository[3], enabling others to benefit from our approach and allowing them to build even more applications.

2 RELATED WORK

This work intersects three major research areas: capacitive sensing, tangibles, and Super-Resolution algorithms. First, we investigate recent developments in capacitive sensing and touch interaction. Then, we discuss tangibles and their feasibility on capacitive touchscreens. Lastly, we explore Super-Resolution algorithms for facilitating tangibles on capacitive touchscreens.

2.1 Tangibles on Capacitive Touchscreens

Grosse-Puppendahl et al. [28] thoroughly covered the large body of research on capacitive touchscreens in HCI. Recently, researchers explored tangibles to improve the lacking interactive experience on touchscreens (e.g., [70, 71]). Tangibles are physical objects, such as pens [16, 70], that serve as input modality with their location and rotation. They were proposed for learning applications [30, 96], music, image and video editing [24, 57, 88], and gaming [4, 6]. For example, GraspDraw [22] allowed users to draw and manipulate geometric primitives, such as lines and rectangles, using two tangible bricks.

Originally, domain-specific detection algorithms tracked embedded fiducial markers, such as ARTag [96] and AR-Toolkit [3] markers with regular (e.g., Pedersen and Hornbæk [66]) or infrared cameras (e.g., Merz et al. [57]). To alleviate the need for additional sensing hardware, tracking gradually shifted towards capacitive touchscreens that offered standalone tracking with a smaller form factor (e.g., [68]). Instead of cameras, touch imprints were used to track spatial point patterns (e.g., [33, 82]) and geometric shapes [70]. Kratz et al. [43], for example, designed tangible knobs with touch point patterns that could be used on Apple iPads. These capacitive tangibles are oftentimes 3D-printed [70, 71] and combine insulating materials with conductive materials to create touch imprints.

However, the coarse sensor size of capacitive touchscreens precludes traditional fiducial markers and their detection algorithms. Cameras can represent areas as small as 2mm and are constantly improving, e.g., in smartphones [88], whereas commodity capacitive touchscreens are limited to 4mm without improvement [47, 56, 76]. Research handles this limitation with bulky tangibles and limits the number of distinguishable objects (e.g., [27, 81]).

To alleviate this issue, Itsy-Bits [70] used deep-learning to classify geometric shapes (12×12mm, n=30) improving upon prior work, for instance, CapCodes (31 × 21mm, n=12) [27]. Still, this approach was limited to a fixed set of tangibles and necessitated elaborate data collection and network training. Steuerlein and Mayer [75] proposed a deep-learning toolkit for simulating and classifying AprilTag markers as small as 24×24mm and geometric shapes. While their toolkit improved upon prior work [70] by reducing data collection effort, their classifier still required network training. Mayer et al. [56]

[1] https://www.tangiplay.com/
[2] https://en.wikipedia.org/wiki/Microsoft_PixelSense
[3] https://github.com/mimuc/super-resolution-for-fiducial-tangibles

argued for a geometric super-resolution algorithm to restore the imprint of fiducial markers and bring back off-the-shelf detection algorithms for AprilTag markers (24×24mm). Yet, this approach required moving tangibles across at least ten images.

2.2 Super-Resolution Algorithms

Super-Resolution (SR) algorithms aim at obtaining a high-resolution (HR) image from one or multiple low-resolution (LR) images [60]. SR goes beyond traditional interpolation algorithms [17, 64], for instance, Lanczos interpolation that does not reconstruct fine details [60]. Nasrollahi and Moeslund [60] contributed an excellent survey on this topic. The most prominent application area is photography (e.g., [15, 89, 93]).

The most common methods are Multi-Image-Super-Resolution (MISR) and Single-Image-Super-Resolution (SISR) [60]. MISR is a reconstruction-based approach aiming to merge multiple LR images into one HR result [60, 78]. As previously mentioned, Mayer et al. [56] proposed a MISR algorithm for capacitive touchscreens. However, prior research highlighted the limitations of MISR [2, 20, 52], such as high computational complexity for an increasing number of frames [20].

SISR is a learning-based approach that aims to reconstruct missing information from a single LR image [60]. The learning is typically achieved by Machine Learning models, such as simple neural networks [32, 59] and Deep Convolutional Neural Networks (DCNN) [14, 39, 40]. SRCNN [15] is a frequently cited example that outperforms state-of-the-art algorithms with a lightweight DCNN. With recent advances in Machine Learning, Generative Adversarial Networks have become attractive for SISR.

2.3 Super-Resolution Generative Adversarial Networks

Generative Adversarial Networks (GANs) learn the distribution of training data to create convincing samples mimicking that distribution [13]. They consist of two networks, the Generator (G) and Discriminator (D), that are trained in competition with each other [13]. Traditionally, GANs were unconditioned and operated on noise vectors from a latent space. These unconditional GANs were used for image [25, 35, 36], 3D shape [90], and audio [53] synthesis.

However, unconditional GANs did not allow direct control over the generated data. For this reason, Mirza and Osindero [58] proposed conditional GANs (cGANs). cGANs opened up new possibilities, such as data augmentation [5], image [73] and speech [42, 65] enhancement, image editing [10], and image style-transfer [34, 50]. As previously mentioned, Steuerlein and Mayer [75] used style transfer to simulate capacitive images from templates of fiducial markers and geometric shapes.

cGANs have also been explored for super-resolving MRI images [9] and photographies [95]. CapContact [76] adopted cGANs to the capacitive image domain. The network mapped LR capacitive images to FTIR HR images of touch points, effectively upsampling the LR image by factor eight [76]. The authors achieved high accuracies (87%) for separating closely adjacent touch points [76]. However, the authors did not pivot their research on tangibles.

2.4 Summary

Tangibles improve the interactive experience of capacitive touchscreens [28]. However, the coarse sensor size precludes the tracking of traditional tangibles equipped with fiducial markers. To alleviate this limitation, researchers explored deep-learning classifiers [70, 75] for a fixed set of tangibles that entailed elaborate data collection and network training. Alternatively, MISR [56] was proposed to restore the imprints of fiducial markers bringing back off-the-shelf fiducial detection algorithms and eliminating the need for neural network training.

In this work, we propose SISR using cGANs. As cGANs were promising for super-resolving touch points [76], we expect them to outperform MISR [56] and super-resolve stationary tangibles from one single image. In contrast to prior work [71, 75], our network is not limited to a fixed number of tangibles and does not require data collection and network training for new fiducial markers. Additionally, we bring back off-the-shelf fiducial detection algorithms and reduce development efforts for custom algorithms.

3 DATA COLLECTION

We super-resolve 30×30 pixel capacitive images of fiducial markers to 60×60 pixel. We define this as a mapping from a low-resolution fiducial marker to a high-resolution counterpart: $f : LR \rightarrow HR$. To train a cGAN on this equation, we require a large dataset of LR and HR image pairs. In this section, we present the selected fiducial markers, outline the apparatus and procedure, and describe the preprocessing steps for the collected data.

3.1 Fiducial Marker Fabrication

We selected four fiducial markers visualized in Figure 2. Our dataset consisted of ten AprilTag 36h11, three AprilTag 16h5, three ArUco original, and three ARToolKit 4×4 markers. This diverse dataset allowed us to assess the generalizability of the network to unseen fiducial markers. All selected fiducial markers can be detected with off-the-shelf detection algorithms, such as ArUco detector [23]. To create $LR \rightarrow HR$ image pairs, we fabricated all fiducial markers in

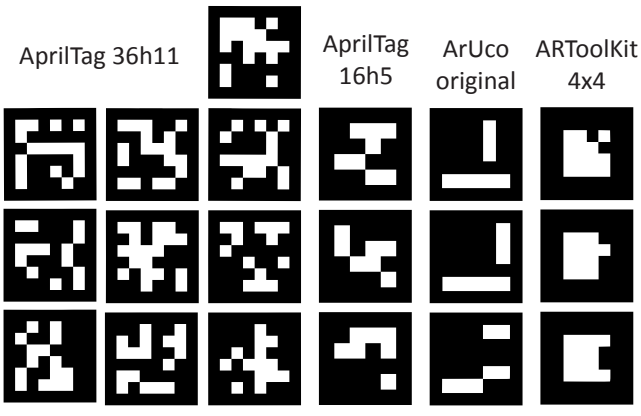


Figure 2: Overview of the selected fiducial markers. Each marker was fabricated for two conditions *SMALL* and *LARGE*. Each condition consists of two resolutions, LR and HR.

two resolutions, LR and HR. For this, the HR markers were twice as large as the LR markers.

Furthermore, we created $LR \rightarrow HR$ marker pairs for two conditions, *SMALL* and *LARGE*. These conditions allowed us to super-resolve fiducial markers with two different pixel sizes and can be expressed as:

$$SMALL : 4\text{mm (LR)} \rightarrow 8\text{mm (HR)} \quad (1)$$

$$LARGE : 6\text{mm (LR)} \rightarrow 12\text{mm (HR)} \quad (2)$$

The pixel size describes the width of the square-shaped black and white pixels encoding information in the fiducial marker. Table 1 shows an overview of the two conditions and the resulting dimensions of the fiducial markers. The marker size was limited by the tablet screen width, which allowed for a maximum diameter of 14.78cm. Therefore, the largest fabricated fiducial marker had a diameter of 13.57cm.

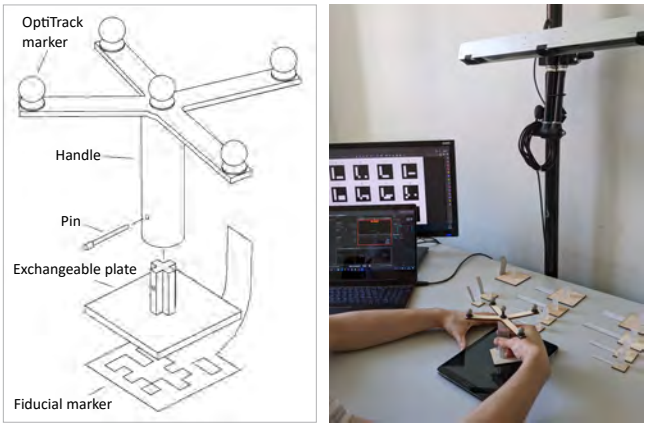
The markers were laser-cut from conductive aluminum-coated paper and attached to 3mm thick wooden plates. We added a strap to each fiducial marker that allowed users to touch the marker and close the electric flow without touching the screen. In total, we fabricated 19 markers × 2 resolutions × 2 conditions = 76 markers.

3.2 Apparatus

We recorded the fiducial markers on a Samsung Galaxy Tab S2 SM-T813 with a 9.7" touch display (2048×1536 pixel) and powered by Android 5.0. Since manufacturers do not grant access to the raw sensor data, we used a custom kernel to record 37×49 pixel (4mm per capacitive pixel) capacitive images at 15fps. We used an OptiTrack-V120:Trio optical motion capture system to track the rotation of the markers on the touchscreen. This allowed us to map LR images to HR images with identical rotation. The device's software Motive was deployed on a Windows laptop and recorded position and rotation at 120fps. We fabricated a custom mount from 3D-printed resin and laser-cut wood to attach five reflective OptiTrack markers required for the optical tracking to the fiducial markers. In the software Motive, we defined the mount as a rigid body and aligned the pivot point with the center of the fiducial markers. Figure 3 visualizes the custom mount and gives an overview of the entire apparatus. As the recording was performed on two devices simultaneously, we relied on Unix timestamps to synchronize the data.

Table 1: Overview of the fabricated fiducial markers. There are two conditions *SMALL* and *LARGE*. Each condition consists of two resolutions, LR and HR. The shape of the fiducial markers is given by border pixels and data pixels. The total tag size ($\sum$) in mm results from the pixel size (mm) and the shape (border pixels, horizontal pixels × vertical pixels).

			SMALL				*LARGE*			
			LR		*HR*		*LR*		*HR*	
Marker type	**n**	**Shape**	**Px**	$\sum$	**Px**	$\sum$	**Px**	$\sum$	**Px**	$\sum$
AprilTag 36h11	10	1, 6×6	4	32	8	64	6	48	12	96
AprilTag 16h5	3	1, 4×4	4	24	8	48	6	36	12	72
ArUco original	3	1, 5×5	4	28	8	56	6	42	12	84
ARToolKit 4×4	3	2, 4×4	4	32	8	64	6	48	12	96



(a) The custom mount. **(b) The setup for data collection.**

Figure 3: (a)The technical sketch of the custom mount used during data collection (left). (b) The setup for data collection.

3.3 Procedure

We attached the tablet with double-sided adhesive tape to a table with a sheet of non-conductive foamed plastic in between to shield it from interference. We fixated the OptiTrack system above the table and calibrated the tablet as the ground plane. The upper left corner of the tablet was the origin of the tracking area. We divided the data collection process into multiple recording sessions and synchronized both devices' local Unix time to an NTP server at the beginning of each session. During the recording sessions, we ground ourselves to the tablet by touching the exposed metal frame. To gather capacitive images in all possible rotations, we steadily rotated each marker clockwise along its y-axis. A recording session for one marker took, on average 4min 22sec (SD: 5sec).

3.4 Preprocessing

Each fiducial marker yielded 4,151 ($SD = 119$) capacitive images and 31,412 ($SD = 577$) OptiTrack samples. In total, we recorded 315,517 capacitive images and 2,387,302 OptiTrack samples. We mapped the rotation recorded by the OptiTrack device to the capacitive images recorded by the tablet using the synchronized Unix timestamps. To account for system latencies, we manually synchronized the first capacitive image containing a marker with the first OptiTrack sample. We corrected the timestamps by an average of 394ms (SD: 698ms). We additionally performed a visual analysis to verify the rotation mapped to the capacitive images manually.

We discarded 8.6% of the data by filtering weak and empty capacitive images with a mean pixel value below 40.0% of the overall mean pixel value. We flipped the remaining 290,609 images to account for the mirrored recording on the touchscreen. Then, we normalized the images and applied contour detection [77] to isolate the blobs of the markers within the capacitive images. To obtain uniform image sizes, we added padding to the blobs. We generated 30×30 pixel images for the LR markers and 60×60 images for the HR markers. Since we required $LR \rightarrow HR$ image pairs, we merged the LR and HR capacitive images by their rotation using an inner join. This merge created 1,581,523 capacitive image pairs. Figure 4

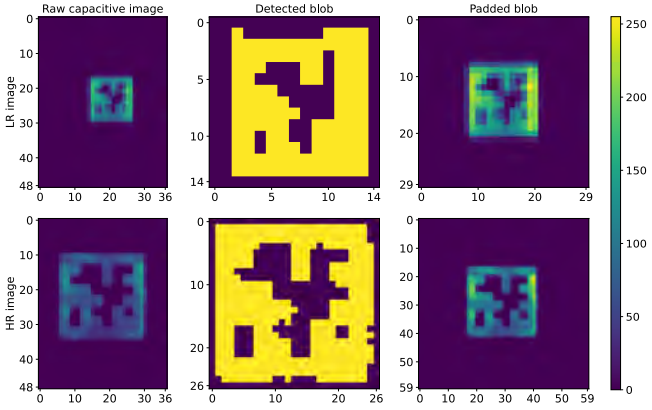


Figure 4: Overview of the preprocessing steps for a $LR \rightarrow HR$ image pair of condition *LARGE*.

visualizes the processing of a $LR \rightarrow HR$ image pair. To further augment the data, we rotated each image pair clockwise by 90°, 180°, and 270°, which quadrupled the size of the data to 6,326,092 samples. We identified the median (MED: 452) of samples per rotation for each pair of markers and balanced the dataset by randomly selecting 452 samples respectively. The balanced dataset contained $452 \times 360 = 162,720$ samples for each pair of markers and 6,326,092 samples in total. Each data sample $\{x_{id}, x_{LR}, x_{HR}, x_r\}$ contained an unique identifier, the LR capacitive image, the HR capacitive image, and the rotation in degrees.

To train a cGAN, we split this data into training, validation, and test datasets using a class-wise split. This method guaranteed unique datasets with no overlapping fiducial markers. We chose eight AprilTags 36h11 markers for the training dataset and the remaining two AprilTags 36h11 markers for the validation dataset. To assess how well the cGAN generalizes to other markers, we reserved all AprilTag 16h5, ArUco original, and ARToolKit 4×4 markers for the test dataset. The training dataset contained 2,603,520 samples, the validation dataset 650,880 samples, and the test dataset 2,928,960 samples. To facilitate training, we scaled the capacitive images to the range $[-1, 1]$. During training, we shifted each image pair by a small random pixel value given by the normal distribution around 0 with a spread of 1. This shift augmented the dataset by adding variance to the capacitive images.

4 SUPER-RESOLUTION NETWORK

Next, we formally define our proposed cGAN. We describe the network's architecture, learning objective, and training process. The presented cGAN is the result of extensive trial-and-error testing and hyperparameter tuning. Additionally, we performed a search to find a suitable model architecture by altering the architecture, e.g., adding or removing layers.

4.1 Definition

Since we condition on the mapping $f : LR \rightarrow HR$, the network is a cGAN. The Generator learned to create fake HR images (henceforth SR images) from given LR images. Therefore, the Generator can be expressed as $G(x_{LR}) \rightarrow x_{SR}$. The Discriminator learned to distinguish between real HR images and SR images. The result y_d describes the probability of the image being real. For real HR images, y_d approximates one, and zero for SR images. This behavior can be expressed as $D(x) \rightarrow y_d$ with $x \in \{LR, SR\}$.

During adversarial training, the Generator and Discriminator compete against each other. The Generator strives to fool the Discriminator with SR images, while the Discriminator strives to recognize all SR images. Literature [34, 75] expressed this learning objective as:

$$G^* = \arg\min_G \max_D \mathcal{L}_{GAN}(G, D) \tag{3}$$

During training, both LR and HR images are available to the cGAN. Once trained, the Generator super-resolves LR images, and the HR images are no longer required.

4.2 Objective Function

The adversarial loss in the learning objective is given by the Discriminator's ability to recognize SR images. Research [34, 49] expressed this as:

$$\mathcal{L}_{GAN}(G, D) = E_{x_{HR}}[\log D(x_{HR})] + E_{x_{LR}}[\log(1 - D(G(x_{LR}))] \tag{4}$$

In addition to the adversarial loss, we employed a pixel-wise L1 loss that encouraged sharper images [34, 49]. Prior work [34, 75] expressed the L1 loss as:

$$\mathcal{L}_{L1}(G) = E_{x_{HR}, x_{LR}}[\| x_{HR} - G(x_{LR}) \|_1] \tag{5}$$

We added the L1 loss with the weighting parameter $w_{l1} = 100$ to the objective function:

$$G^* = \arg\min_G \max_D \mathcal{L}_{GAN}(G, D) + w_{l1} \cdot \mathcal{L}_{L1}(G) \tag{6}$$

4.3 Generator Architecture

The Generator has 1,002,433 parameters. Its architecture draws inspiration from prior work on SISR [49, 76]. Furthermore, we adhered to the architectural recommendations by Radford et al. [67]. As depicted in Figure 5a, the network consists of four residual blocks [29] with convolutional layers. Residual blocks perform well in SISR with a reasonable number of parameters [49, 76]. Residual blocks apply an identity mapping by adding the output of the block to its input [29]. The PixelShuffler layer [72] transforms 30×30 pixel images into 60×60 pixel images. The final layer uses a *tanh* activation function to scale the 60×60 SR images to the initial range $[-1, 1]$.

4.4 Discriminator Architecture

The Discriminator has 1,187,073 parameters. It fuses prior work on SISR [49, 76] with the PatchGAN [34, 50] that counteracts blurry images, similar to Steuerlein and Mayer [75]. Again, we adhered to the architectural recommendations by Radford et al. [67]. It's architecture is illustrated in Figure 5b. The Discriminator downsamples the images in six convolutional blocks using strided convolutions [74]. The final layer uses a Sigmoid activation function to scale the 8×8 pixel patches to the probability range $[0, 1]$.

4.5 Adversarial Training

Standard backpropagation [69] adjusted weights and biases of the cGAN to minimize the objective function. We trained the cGAN

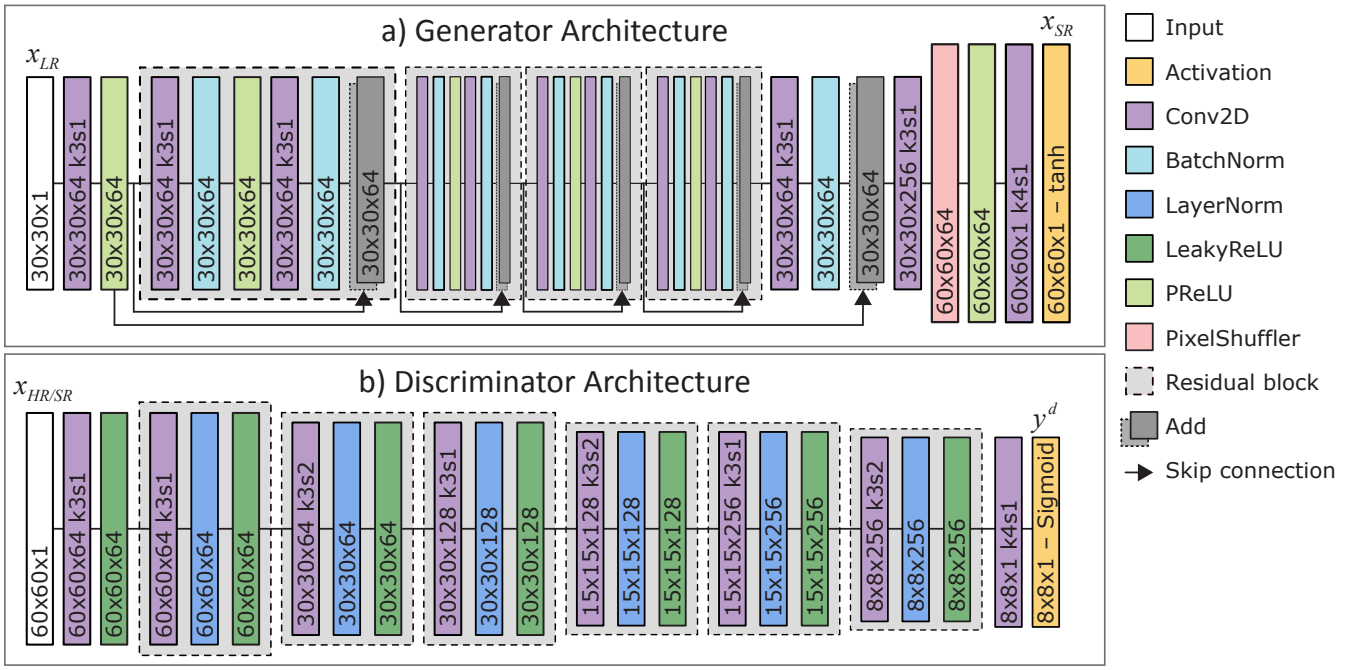


Figure 5: a) The architecture of the Generator with 1,002,433 parameters. b) The architecture of the Discriminator with 1,187,073 parameters.

using the Adam optimizer [41]. The Generator had the learning rate $lr_G = 4 \times 10^{-4}$, and the Discriminator $lr_D = 2 \times 10^{-4}$. We trained the network for 4520 epochs. This took 25 hours and 13 min on an Nvidia Tesla V100 GPU. Each epoch contained 180 training and 45 validation batches with a batch size of 32. Statistically, one sample for each marker, condition (*SMALL* and *LARGE*), and rotation occurred within one epoch. We saved preliminary networks every 100 epochs. We assessed the image quality and detection accuracy of all preliminary networks for the validation and test datasets. Based on this assessment, we selected the best network that trained for 500 epochs.

5 EVALUATION

In this section, we evaluate the SR images created by the GAN. For this, we froze the Generator and converted it to the Keras H5 (.h5) format. First, we assess the image quality visually and with various metrics proposed by prior work. Then, we investigate the network's ability to enable off-the-shelf fiducial detection algorithms to track fiducial markers reliably. For this, we use the AprilTag [85], ArUco [23], and ARToolKit[4] detectors. Before the evaluation, we performed a grid search combined with trial-and-error to determine the best detection parameters.

5.1 Image Quality

We assessed the quality of the generated SR images visually. For this, we de-scaled the capacitive images to the range $[0, 255]$. We chose Lanczos-4 interpolation [45] as a baseline for the image quality, as it has been shown to yield the best results among interpolation algorithms [64]. Figure 6 shows randomly selected fiducial markers of each type. Compared to the Lanczos-4 baseline, the SR images resemble the HR images better. Particularly, the SR images of *SMALL* markers outperform the baseline.

In line with prior work [49, 94], we also assessed the metrics MAE, SSIM, and PNSR. The pixel-wise Mean Absolute Error (MAE) was part of the objective function for the GAN expressed as $\mathcal{L}_{L1}(G)$. It described the pixel-wise error between real and fake images. The Structural Similarity Index (SSIM) approximates the perceived image quality as a value between 0 and 1, where 1 describes identical images. We used the function parameters proposed by Wang et al. [87]. The Peak-Signal-to-Noise Ration (PSNR) approximates the reconstruction quality in dB. Higher values indicate better image quality. For this, we also used the pre-trained Tensorflow implementation of the ESRGAN [5] proposed by Wang et al. [86] as an additional SISR baseline. Table 2 shows all performed analyses.

Since most pixels had the value zero, larger markers with more non-zero pixels lead to larger errors. Overall, our SR images outperformed traditional Lanczos-4 interpolation and the ESRGAN baseline for all metrics; see Table 2. The largest MAE for SR images was 6.99, which meant a small 2.74% pixel-wise discrepancy. The largest MAE for the Lanczos-4 baseline was 11.21. Therefore, the pixel-wise discrepancy was 4.40%. The ESRGAN baseline performed similarly to the Lanczos-4 baseline without noticeable improvements.

Lastly, we compared the distribution of pixel values between HR and SR images. Figure 7 visualizes histograms for each dataset. The SR images from the validation dataset deviated moderately from the HR images for pixel values 90-140. This deviation increased

[4]https://github.com/artoolkitx/jsartoolkit5

[5]https://tfhub.dev/captain-pool/esrgan-tf2/1

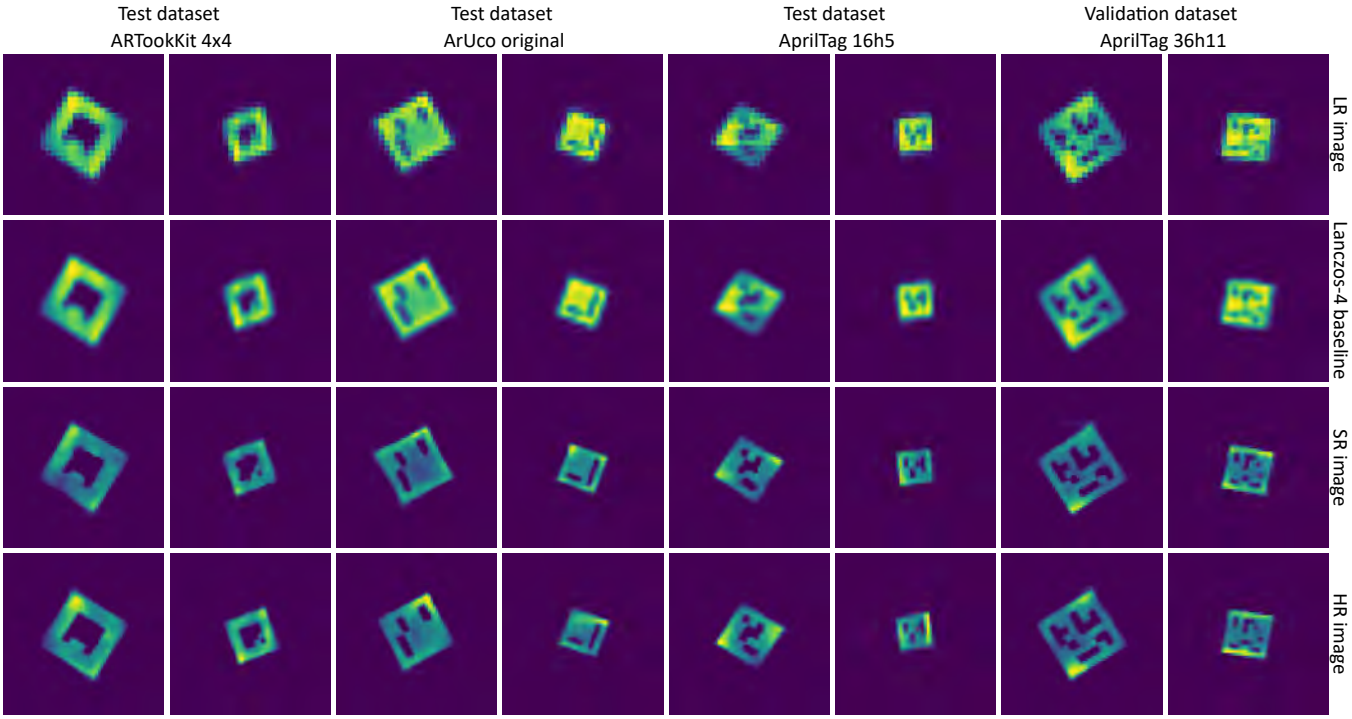


Figure 6: The *SMALL* and *LARGE* markers from the validation and test datasets are shown in paired columns, the *LARGE* ones on the left and the *SMALL* ones on the right. The first show shows representative ground truth images for each of the eight different subsets of the dataset. Recorded LR images upsampled using Lanczos-4 interpolation [45] are shown for comparison in the second row. The SR images in the third row are created using our Generator. Finally, a corresponding ground truth high-resolution image is shown in the last row.

Table 2: MAE, SSIM and PSNR for SR images created by the network compared to the real HR images. The baselines are Lanczos-4 interpolation [45] using the OpenCV implementation and the Tensorflow implementation of the ESRGAN [86]. The optimal values are MAE = 0, SSIM = 1 and PSNR = ∞. The values in green indicate best results.

	Our						**Lanczos-4 baseline [45]**						**ESRGAN baseline** [86]					
	MAE		**SSIM**		**PSNR**		**MAE**		**SSIM**		**PSNR**		**MAE**		**SSIM**		**PSNR**	
	M	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**
Training datasets																		
AprilTag 36h11 - *SMALL*	2.86	.34	.93	.05	25.9	3.6	5.7	.47	.89	.03	21.2	1.5	5.81	.39	.88	.04	20.7	1.7
AprilTag 36h11 - *LARGE*	4.03	.17	.88	.07	24.8	3.1	11.2	.41	.8	.05	18.3	1.5	11.15	.38	.8	.06	18.	1.6
Validation datasets																		
AprilTag 36h11 - *SMALL*	3.	.	.91	.05	24.8	2.6	6.79	.4	.87	.03	20.	1.3	6.88	.33	.87	.04	19.5	1.5
AprilTag 36h11 - *LARGE*	5.92	.27	.86	.07	22.8	2.3	11.02	.12	.81	.05	18.5	1.4	11.	.25	.8	.06	18.1	1.5
Test datasets																		
AprilTag 16h5 - *SMALL*	2.	.02	.93	.03	25.3	2.7	3.03	.17	.92	.02	23.1	1.6	3.29	.45	.92	.03	22.6	1.9
AprilTag 16h5 - *LARGE*	4.02	.14	.89	.04	23.1	2.3	6.	.06	.88	.04	21.1	1.5	6.	.1	.88	.04	20.8	1.8
ArUco original - *SMALL*	4.	.	.89	.04	22.3	2.	5.71	.45	.88	.03	20.5	1.4	5.9	.31	.87	.03	20.	1.6
ArUco original - *LARGE*	6.07	.25	.84	.05	21.5	2.1	10.3	.46	.8	.05	18.4	1.5	10.11	.33	.81	.05	18.2	1.6
ARToolKit 4×4 - *SMALL*	3.11	.32	.92	.04	24.	2.2	5.01	.09	.89	.03	21.7	1.8	5.01	.11	.9	.04	21.5	1.9
ARToolKit 4×4 - *LARGE*	6.99	.09	.83	.05	21.3	2.	11.02	.3	.8	.06	18.6	2.1	10.69	.52	.8	.06	18.4	2.2

Table 3: Marker detection for the SR images created by the network. The shape of the fiducial markers is (border pixels, horizontal pixels × vertical pixels). The baselines are Lanczos-4 interpolation [45] using the OpenCV implementation and the pre-trained Tensorflow implementation of the ESRGAN [86]. The rotation MAE is relative to the recorded OptiTrack rotation. The values in green indicate best results.

			Our			Lanczos-4 baseline [45]			ESRGAN baseline [86]		
			Pred.	Rotation		Pred.	Rotation		Pred.	Rotation	
	Markers	Shape	Acc.	MAE	SD	Acc.	MAE	SD	Acc.	MAE	SD
Training datasets											
AprilTag 36h11 - *SMALL*	587	1, 6x6	97.7	2.3	7.6	.	–	–	.3	2.6	7.7
AprilTag 36h11 - *LARGE*	587	1, 6x6	75.1	1.6	5.1	74.7	2.1	6.	14.7	2.5	7.6
Validation datasets											
AprilTag 36h11 - *SMALL*	587	1, 6x6	67.8	2.1	7.5	.	–	–	.3	2.7	8.
AprilTag 36h11 - *LARGE*	587	1, 6x6	96.3	1.6	5.6	77.9	1.9	5.4	17.5	2.2	6.5
Test datasets											
AprilTag 16h5 - *SMALL*	3	1, 4x4	91.9	3.9	10.2	.	26.7	27.2	2.7	4.8	11.6
AprilTag 16h5 - *LARGE*	3	1, 4x4	99.3	2.6	7.4	93.7	3.2	6.9	55.6	2.7	5.
ArUco original - *SMALL*	1024	1, 5x5	58.	4.8	9.3	.	–	–	1.2	2.	7.9
ArUco original - *LARGE*	1024	1, 5x5	67.6	2.1	6.1	25.8	2.	3.6	71.9	2.2	5.4
ARToolKit 4×4 - *SMALL*	5	2, 4x4	20.3	3.	2.6	6.6	3.8	1.7	2.6	1.9	1.6
ARToolKit 4×4 - *LARGE*	5	2, 4x4	85.7	3.3	2.8	35.9	2.9	2.3	2.9	1.1	1.6

for SR images from the test datasets. The ARToolKit 4×4 markers showed the largest deviation for pixel values 60-140.

5.2 Fiducial Marker Detection

We assessed the network's ability to super-resolve fiducial markers based on the detection accuracy and the rotation MAE, Table 3 shows the detection results. The recorded OptiTrack rotation served as the ground truth for the rotation MAE. Before detection, we descaled the images to the range $[0, 255]$ and normalized them. Then, we upsampled the SR images to 600×600 pixels using Lanczos-4 interpolation. We applied Gaussian blur with a 5×5 kernel and thresholded the images using Otsu's method [63]. This postprocessing is done to support the detection algorithms and improve the results.

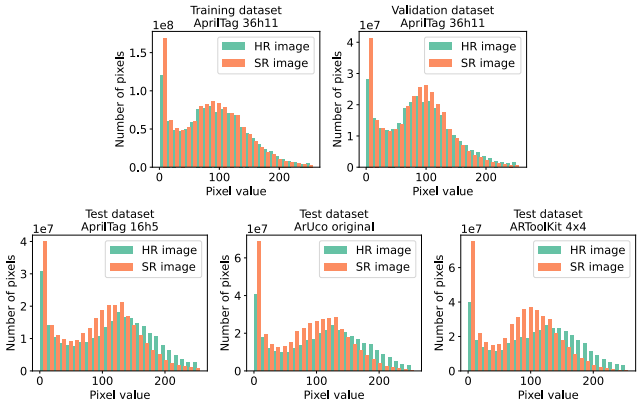


Figure 7: Distribution of pixel values for the training, validation, and test datasets. Pixels with a value of 0 are not visualized. The bin size is 10.

The *LARGE* markers have higher detection accuracy than the *SMALL* markers for the validation and test datasets. The largest discrepancy occurred for the ARToolKit 4×4 markers. Here, the detector was accurate for 85.7% of *LARGE* markers and only 20.3% of *SMALL* markers. Yet, there is no statistically significant difference in detection accuracy ($t(6) = 1.22$, $p = 0.29$) and rotation MAE ($t(6) = 1.93$, $p = 0.13$) between all *LARGE* and *SMALL* markers. The largest MAE was 4.83, which meant a small 1.3% deviation from the ground truth.

We used Lanczos-4 interpolation [45] from OpenCV as a simple baseline, and a pre-trained ESRGAN proposed by Wang et al. [86] as an advanced SISR baseline. For the Lanczos-4 baseline, the LR images were upsampled directly to 600×600 pixels, and the same postprocessing was applied. Overall, our SR images considerably outperformed both baselines, especially for *SMALL* markers. The detection accuracy of *SMALL* markers was 67.16% compared to only 1.85% for the Lanczos-4 baseline. The best improvement for *LARGE* markers was from 35.9% to 85.0% detection accuracy for the ARToolKit 4×4 markers. Notably, the ESRGAN baseline outperformed the SR images for *LARGE* ArUco original markers. Otherwise, the ESRGAN baseline underperformed both our SR model and the Lanczos-4 baseline.

5.3 Comparison to Mayer et al.'s [56] approach

As a next step, we compare our result to the geometric MISR method by Mayer et al. [56]. Our test data set using their pipeline achieves an accuracy of 82.8% for SMALL 16h5 AprilTags and 99.9% for LARGE 16h5 AprilTags.

Mayer et al.'s [56] method rotates the 37 input images and aligns them. As the user additionally might rotate the tangible during input, there is no single unique rotation making it impossible to determine a single ground truth value. For our dataset, within the 37 input images, the rotation varied between 3.8° and 49.8° ($M = 11.1°$, $Med = 3.8°$, $SD = 3.8°$). For the performance measure of the approach on our dataset, we used the average rotation over the 37 images; however, the results will be skewed due to the variation in rotation. The average MAE is 9.8° for the SMALL 16h5 AprilTags ($SD = 25.1°$, $min = 0.0°$, $max = 171.5°$) and for the LARGE 16h5 AprilTags the MAE is 8.3° ($SD = 21.2°$, $min = 0.0°$, $max = 179.5°$).

5.4 User Evaluation

After evaluating the quality and comparing the system to two baselines, we investigated the detection accuracy and performance in a real-world setting. Here, we asked participants to place fiducial markers on our tablet that super-resolved the capacitive images and performed the detection.

Apparatus. For this evaluation, we used five *SMALL* and five *LARGE* AprilTag 36h11 and 16h5 markers, for a total of 5 markers × 2 types × 2 conditions = 20 markers. We fabricated additional markers to ensure that all markers were unseen by our model. As a device, we used the same Samsung Galaxy Tab S2 SM-T813.

Procedure. Upon arrival, we briefed the participants about the study, answered any open questions, and then asked them to sign an informed consent form. Next, we asked them to place a marker on the tablet and wait until the tablet prompted them to continue with the next marker. After they placed each fiducial marker three times on the tablet, we thanked them for participating. This procedure yielded 60 samples of real-world tangible interaction for each participant.

Participants. We recruited 11 participants (three female, eight male) with an age range between 25 and 63 ($M = 31.5$, $SD = 10.2$) to participate in the study. Our study took approximately 10 minutes. All participants volunteered to take part in the study.

Evaluation Results. Table 4 shows the results of the user evaluation. All markers were detected with a high accuracy of approximately 100%. However, the detection of the AprilTag 16h5 markers took 806ms, while the detection of the AprilTag 36h11 markers took 1937ms on average. Similarly, the standard deviation for the AprilTag 36h11 was larger (3000ms). For example, the *LARGE* AprilTag 36h11 marker with ID 66 took 4418ms to detect, while ID 54 took 832ms. Also, as the number of *Detection Attempts* varies between 2 and 7, the first detection attempt failed.

6 APPLICATIONS AND DEPLOYMENT

In this section, we deploy the Generator on a commodity tablet for real-time fiducial tracking. We illustrate three showcases where small tangibles improve the interactive experience. Two additional showcases highlight the potential of super-resolved fiducial markers for security and authentication.

Table 4: Results of the real-world user study. We show the *Accuracy*, the *Time to Detection* (ms), and *Detection Attempts* (frames) indicate the duration between a marker's first contact with the tablet screen to the first detection result.

		Time		Attempts	
	Acc.	**M**	**SD**	**M**	**SD**
AprilTag 36h11 - *SMALL*	100	2,231	2,748	7.	8.4
AprilTag 36h11 - *LARGE*	100	1,642	3,251	5.2	11.4
AprilTag 16h5 - *SMALL*	98.8	869	905	2.4	.9
AprilTag 16h5 - *LARGE*	100	743	270	2.3	.6

6.1 Mobile Deployment

We deployed the Generator on the Samsung Galaxy Tab S2 SM-T813, which was previously used for recording the data with the custom kernel. For this, we froze the Generator and converted it to the TensorFlow Lite (.tflite) format[6], which can be used for on-device inference. After the conversion, the Generator shrunk from 11.6MB to 3.8MB. Furthermore, we used the Android implementation of the AprilTag detector[7] and the OpenCV library[8].

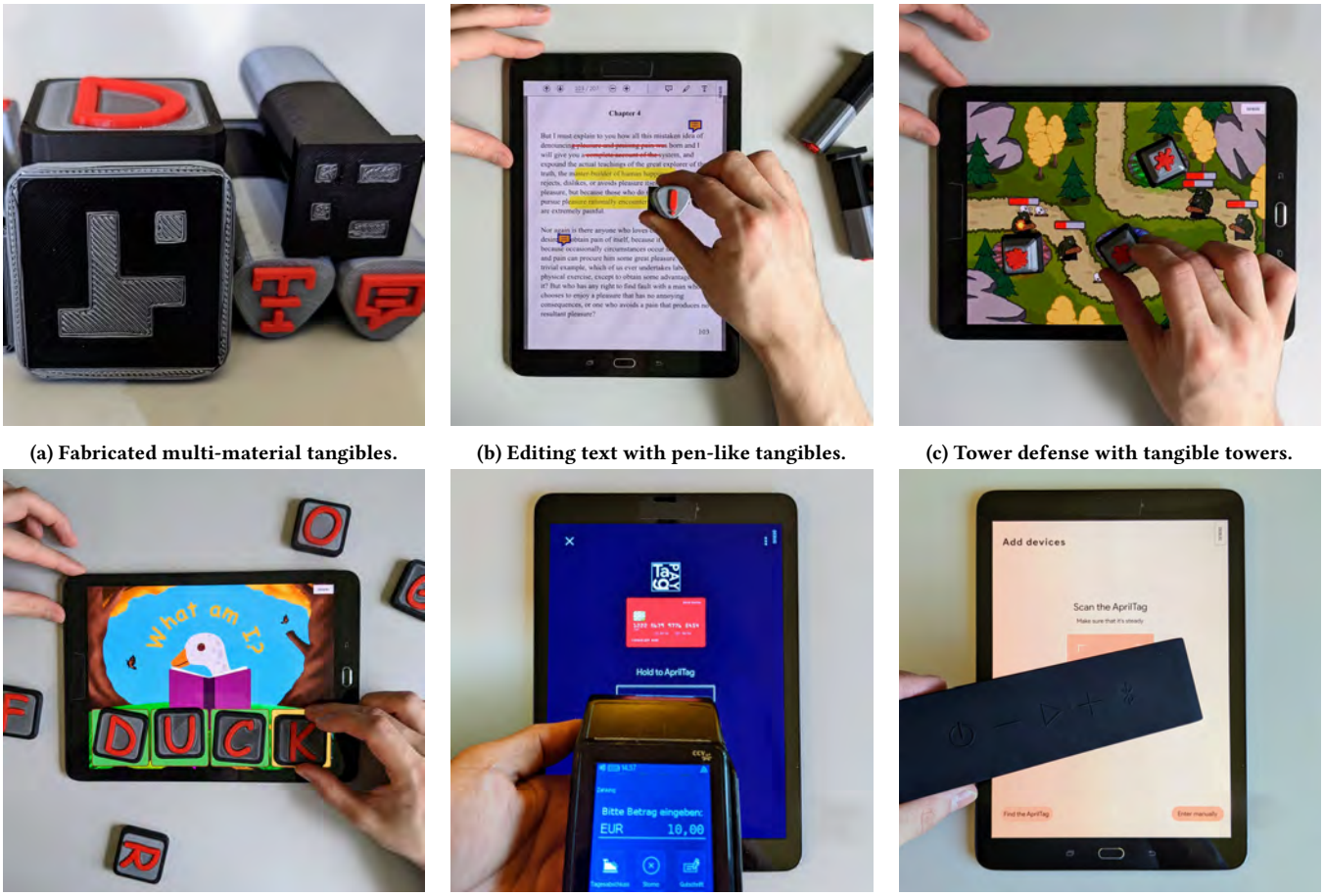
As mobile devices have limited processing power, we adjusted the postprocessing of the SR images. Here, we upsampled the SR images to 200×200 pixel using Bicubic interpolation with a 4×4 kernel. We also omitted the Gaussian blur before Otsu's thresholding. Across all AprilTag markers, the detection accuracy was only 0.80% lower and the rotation MAE was 0.21 larger. This small loss allows for a significant performance gain.

We developed a benchmark to assess the duration of the individual steps from recording the capacitive image to the final detection results. We performed the benchmark on the Samsung Galaxy Tab S2 SM-T813, which features a Qualcomm Snapdragon 620 processor and on the Samsung Galaxy S21 5G SM-G991, which features a significantly faster Exynos 2100 processor for comparison. Since the Samsung Galaxy S21 5G SM-G991 does not support the custom kernel necessary for accessing the capacitive images, we processed random noise. Processing random noise instead of capacitive images allowed us to assess the performance on modern devices that are yet to receive a custom kernel. Table 5 show the results of the benchmark averaged over 1000 runs. On the Samsung Galaxy Tab S2 SM-T813, all steps took 322ms in total (3fps). For the showcases, 3fps was sufficient. On the faster device, the duration sunk to 150ms.

6.2 Conductive Tangibles

For the showcases, we aimed at high-fidelity tangibles as opposed to the low-fidelity prototypes used for recording the data. We designed and 3D-printed multiple tangibles with a combination of conductive and non-conductive materials, as proposed by prior work [54, 70]. We used black Protopasta Composite PLA filament for the conductive core with a volume resistivity of 30–115 Ω×cm. For the non-conductive parts, we used regular PLA filament in the colors gray and red. Similar to Schmitz et al. [70], we used a Prusa

[6] https://www.tensorflow.org/lite
[7] https://github.com/johnjwang/apriltag-android
[8] https://opencv.org/android/



(a) Fabricated multi-material tangibles.

(b) Editing text with pen-like tangibles.

(c) Tower defense with tangible towers.

(d) Literacy learning with tangible letters.

(e) Fiducial marker to increase payment security.

(f) Scanning a fiducial marker instead of a QR-code to connect smart devices.

Figure 8: (a) Conductive tangles, the red and grey parts are non-conductive PLA filaments, and the black core is conductive PLA filament. (b-f) Five showcases where tangibles improve the interactive experience on capacitive touchscreens.

MK3 3D printer with the Mosaic Palette 3 Pro multi-material extension. To improve conductivity, we increased the infill density to 50% (default: 15%) and the transition length to 330mm (default: 105mm). We assigned the materials and sliced the models with the Mosaic online tool Canvas3D[9]. Figure 8a shows the fabricated tangibles.

6.3 Example Applications

We created five example applications shown in Figure 8 in which we used the super-resolution model to facilitate enhanced input on commodity capacitive screens. We implemented all applications on a Galaxy Tab S2 SM-T813. We now summarize the examples; see Video Figure.

Text Editing. Workplace culture has shifted towards mobile work [8]; thus, many people now depend on mobile devices to perform their office tasks marking work cumbersome. Inspired by literature [16, 70], we prototyped a text editing showcase that reduces the number of inputs by mapping digital tools to familiar analog pens. Our showcase features three core functions frequently used in editing: highlight, strike, and comment. Upon touching the tablet with a pen, the respective tool is highlighted in the menu and can be used accordingly. The highlighting pen adds a transparent yellow overlay, and the strike pen a solid red dash when moving the tangibles along a line of text. The comment pen adds a comment bubble at the selected position in the text. This concept generalizes to applications, where frequent switching between tools is necessary, such as digital painting, 3D modeling, and video editing [57].

Mobile Gaming. Tangibles bring physicality to mobile games [1, 4, 81], enhance the players' enjoyment [1, 6] and interest [6]. We created a tangible experience for the well-known tower defense genre. The goal is to build defenses, traditionally towers in strategic positions, and destroy all approaching enemies. Here, the player places three towers (with unique powers: fire, ice, and poison) as a defense on the tablet. Once all towers are placed, enemy orcs approach, and the towers launch damaging projectiles. We chose the *SMALL* AprilTag 16h5 markers, which leaves room for more types

[9]https://canvas3d.io/

Table 5: Detection time (ms) of the individual processing steps averaged over 1000 runs on two Samsung Galaxy devices. For comparison, the duration using Lanczos-4 interpolation and Mayer et al.'s [56] approach are given.

	M	SD	Min	Max
Samsung S2 SM-T813				
Preprocessing	3	1	1	21
Inference	124	14	108	154
Postprocessing	92	13	79	127
Detection	101	13	84	176
Total	320	41	276	433
Samsung S2 SM-T813 using Lanczos-4 baseline [45]				
Preprocessing	2	1	1	19
Interpolation	27	10	8	43
Postprocessing	106	13	77	140
Detection	90	4	85	159
Total	225	24	174	311
Samsung S21 5G SM-G991				
Preprocessing	–	–	–	–
Inference	31	13	15	84
Postprocessing	3	2	2	36
Detection	87	7	55	99
Total	121	17	95	211
Mayer et al. [56] using MISR				
Total	2,500	–	–	–

of defensive towers and upgrades. This concept is inapplicable to other genres, especially table-top-inspired games with a top-down view (e.g., [79, 80]).

Literacy Learning. By enabling tangibles on today's touchscreen, we further support a wide range of tangible learning, see the literature review by Li et al. [51]. Inspired by Fan et al. [18, 19], we created a tangible spelling playground, where children learn to spell words with tangible letters. The child selects and places the letters correctly on the tablet. The 30 different tags of this tag family support the full English alphabet and leave room for extensions, such as punctuation.

Payment Security. NFC payments are ubiquitous, yet to improve security, researchers have proposed vibration as an additional layer of security [12, 46, 91]; however, these moving parts are subject to wear and tear. Thus, we compared a tag into the payment terminal to be recognized by mobile devices, allowing for secure payment. Additionally, due to the direct contact, the interaction is explicit in nature, offering orthogonal security to wireless transactions such as NFC.

Smart Home. Smart home appliances need unique identifiers. Today, we see ugly QR codes stuck on devices. In line with Mayer et al. [56], we support tangibles with a high payload while also being invisible, hidden in the material of the device, cf. Schmitz et al. [70]. This allows the user to scan the conductive tag on the touchscreen when connecting new smart devices while being hidden most of the time when the identifier is not needed.

7 DISCUSSION

We developed a Super-Resolution Generative Adversarial Network to super-resolve fiducials and facilitate tangibles on capacitive touchscreens using off-the-shelf detection algorithms. Our approach builds upon prior work on super-resolving capacitive images [56, 75, 76]. When we compare our results to Mayer et al.'s [56] approach, we achieve similar results in a fraction of the time. Mayer et al.'s [56] approach need to wait for 2466ms (37 images) plus processing without rotation estimation; in contrast, our approach takes 322ms on the same device allowing for a voting process to improve the accuracy further. In contrast to other deep-learning approaches that facilitate tangibles, we are not limited to a fixed number of distinguishable objects [70] and do not require data collection and network training for new fiducial markers [70, 75]. As such, we argue that our approach achieves comparable results without requiring training any further deep learning model. We achieve this by enabling off-the-shelf fiducial detection algorithms to operate on capacitive images. Thus, we reduce the development effort for customs to enable tangibles to be used on touchscreens.

The network generalizes well to unseen AprilTags, ArUco, and ARToolKit markers that can be tracked with off-the-shelf detection algorithms. We argue that it generalizes beyond this set and can super-resolve other square-shaped fiducials, such as ChiliTag[10] and ARTag [21] markers. The network achieved promising results for image quality and detection accuracy. The pixel-wise discrepancy of the smallest markers in our dataset (24×24mm) was only 0.8%. We were able to accurately detect these markers with a high accuracy of 91.9% and a small rotation MAE of 3.85. Overall, the image quality and detection accuracy were considerably better than the Lanczos-4 interpolation baseline. Moreover, our model outperformed the pre-trained ESRGAN baseline in all but one case for accuracy; however, the rotation MAE provided by the baseline was better for 50% of the test sets, but only by $\sim 2°$. Re-training the ESRGAN model to better support the fiducial structure could result in high even higher performance.

Furthermore, we deployed the network on a commodity mobile device for real-time fiducial tracking. We presented three showcases that improve the interactive experience with tangibles and two showcases that highlight the potential of super-resolved fiducial markers for security and authentication. In this section, we discuss several aspects regarding fiducial type and size, thresholding, overfitting, and mobile performance.

7.1 Adaptive Thresholding

We observed difficulties with the adaptive thresholding of capacitive images. The detection accuracy of *LARGE* markers (75.1%) was lower than the accuracy of *SMALL* markers (97.7%) in the training dataset. This anomaly is likely attributed to Otsu's adaptive thresholding that was applied to the images before detection. Otsu's method adapts the thresholding value to the images' histograms instead of using a fixed value. As can be seen in Figure 9, the *LARGE* markers contain areas with low pixel values that are omitted during

[10] https://github.com/chili-epfl/chilitags

Figure 9: Thresholding error of a *LARGE* AprilTag 36h11 marker. Otsu's adaptive thresholding omits border pixels precluding detection. Manual thresholding at pixel value 40 yields correct results.

thresholding. This problem occurs since large touch areas, such as palms or large fiducials are less accentuated than small touch areas, such as finger touchpoints [76]. This issue highlights the need for a domain-specific thresholding method that adapts well to different touch area sizes.

7.2 Overfitting to Shape

The discrepancy between the MAE of *LARGE* training (4.03) and validation (5.92) markers suggested overfitting to the training dataset. The training dataset contained AprilTag 36h11 markers only, with one border pixel and 6×6 data pixels. Despite this overfitting, the accuracy of the training dataset (86.4%) was lower than the accuracy of the AprilTag 15h6 markers in the test dataset (95.6%). AprilTag 15h6 markers also have one border pixel but only 4×4 data pixels. This suggests that the network did not overfit to the number of data pixels.

Alternatively, the ARToolKit 4×4 markers have 4×4 data pixels and two border pixels. They score the lowest detection accuracy (53.0%). Therefore, the network is likely overfitting to the number of border pixels. This issue can be alleviated with more diverse training data in the future. Simultaneously, it does highlight the generalizability of the network towards variable data pixel shapes.

7.3 Fiducial Marker Size Effect

Intuitively, yet not statistically significant, small fiducials are harder to detect than large fiducials. For all test datasets, the *LARGE* markers (84.2%) were detected more accurately than *SMALL* markers (56.7%). This also applied to the rotation MAE of the *LARGE* (2.67) and *SMALL* markers (3.89). This can be attributed to the pixel size of the fiducial markers. *LARGE* markers have a pixel size of 6mm, which is 50% larger than the *SMALL* markers (4mm). Taking the border pixels and data pixels into account, the *SMALL* ARToolKit 4×4 markers, for instance, have a total size of 32×32mm and cover an area of 1,024mm^2. The LARGE markers of the same type have a total size of 48×48mm and cover a considerably larger area of 2,304mm^2. Since capacitive touchscreens have a limited sensor size of 4mm, the pixel size heavily affects the detection accuracy. While *SMALL* AprilTag 16h5 markers achieved a high accuracy (91.9%), other fiducials, such as ARToolKit 4×4 (20.3%) were less accurate. Nonetheless, the SR images created by the network significantly outperformed the baseline. This illustrates the potential of SR for capacitive images and leaves room for improvement in future work, for example, collecting HR images with higher resolution on a larger capacitive touchscreen.

7.4 Fiducial Marker Type Effect

The type of fiducial also impacts the detection accuracy. Some fiducials guarantee a minimum Hamming distance between similar markers. AprilTag 36h11 markers, for instance, have a Hamming distance $d = 11$, which means that the detector can detect $d/2$ and correct $(d-1)/2$ pixel errors [62]. ArUco original and ARToolKit 4×4 markers do not have a minimum Hamming distance. This means the detector cannot detect and correct pixel errors. Therefore, the detection accuracy for fiducials without a minimum Hamming distance is lower. However, these fiducials encode a larger number of markers. AprilTag 16h5 markers encode 30 markers with 4×4 pixel data, while ARToolKit 4×4 markers encode 50 markers. This results in a trade-off between detection accuracy and the number of markers.

Furthermore, the baseline accuracies suggest that the detection algorithms do not perform equally well on capacitive images. The AprilTag detector achieves a baseline accuracy of 76.3% for LARGE AprilTag36h11 markers. The ARToolKit detector scores only 25.8% for LARGE ARToolKit 4×4 markers. Although the small number of ARToolKit 4×4 markers (50) compared to AprilTag 36h11 markers (587), the ARToolKit detector performs poorly on capacitive images. These insights highlight the need for careful fiducial selection when designing tangible applications on capacitive touchscreens.

7.5 Fiducial Marker Fabrication and Data Collection

Additionally, the network is influenced by the precision of the fabricated fiducial markers and the data collection. Despite laser-cutting, small errors, for instance, evaporated material, can lead to deviations from the desired shape, especially for small markers. However, 3D-printing tangibles with conductive and non-conductive materials allow sub-millimeter precision [54, 70]. Since this is time and material-consuming, we only used 3D-printed tangibles for our showcases. With advancing printing technology, we aim to use accurate 3D-printed tangibles for data collection in the near future.

7.6 Real-World User Evaluation

Our user evaluation provided further insights into real-world tangible interaction, for example, the time to detection. Some markers, such as ID 54 can be detected approximately five times faster than ID 66, despite both being *LARGE* AprilTag 36h11 markers. This indicates that each marker's pixel pattern impacts detection. Some patterns are better suited for a low-resolution capacitive touchscreen than others. This highlights the need for careful fiducial selection when designing tangible applications. We performed a visual inspection of the two markers. In contrast to our first assumption, we found that the marker with the lower detection time has more unconnected data pixels. We assumed that more unconnected data pixels would be harder to detect. Additional in-depth investigations are needed to answer to the question if the difference in detection time is due to the physical build and layout of the tag or due to the model.

Furthermore, the user interaction manifests itself in the detection times and frames until a successful tag detection. We note that the system took a minimum of two frames until detection. This is because the capacitive image generated by the first contact with the capacitive screen is incomplete and, therefore, unsuitable for detection. This issue can be addressed in tangible applications by performing continuous detections or applying bounding box algorithms to discard incomplete images. Moreover, when the users place the marker slowly, there is a higher potential for incomplete marker imprints on the sensor. We hypothesize that one major reason the 36h11 markers are slower to be recognized is that they take more time to place on the screen by the user properly as they are physically bigger.

7.7 Mobile Performance

The long detection times on our older Samsung Galaxy Tab S2 SM-T813 from the user evaluation rained concerns if our model is suitable to support tangible interaction. Therefore, we tried to contextualize the detection times better by using a new device (Samsung S21 5G SM-G99) and measuring individual steps of the process.

Today, all high-quality detection methods that use the capacitive image depend on a custom kernel to access the raw capacitive images, e.g., [11, 48, 55, 92]. This access is slow, achieving only about $\sim 15fps$, and taxing the debug interface even more, would slow down the Android UI update rate making interactions impossible. We note that the UI update rate and the pull loop for the capacitive sensor are not the same, 60fps screen update smartphones do not necessarily have 60fps touch updates. However, we argue that with increased demand for direct access in the research community [56, 71, 75], manufacturers will provide fast access in the future.

We need to take the detection time apart to understand the remaining cost. On our older test device, pre-processing, post-processing, and detecting capacitive markers take a considerable amount of time (196ms of 322ms). However, the inference, running the model, takes only 124ms. Modern smartphones and tablets facilitate fast inference (31ms using the Samsung S21 5G SM-G991) through accelerated Tensorflow Lite networks[11]. Upsampling the SR images to 200×200 pixel can also be accelerated with faster processors. Yet, the AprilTag detector improves only marginally on modern devices and still takes 87ms. This issue can be alleviated by performing one initial detection and using the bounding box and feature-matching algorithms to track the rotation in real time.

In summary, our benchmark revealed that the bottleneck is not the network (inference time) but the detection algorithms and the processing around the model. We argue that with more engineering work, this can be improved in the future, but this was not the focus of this work.

8 CONCLUSION

We proposed a Super-Resolution Generative Adversarial Network to super-resolve fiducials and facilitate tangibles on capacitive touchscreens using off-the-shelf detection algorithms. The network super-resolves 30×30 pixel capacitive images of fiducial markers to 60×60 pixel outperforming traditional interpolation algorithms. This enabled off-the-shelf fiducial detection algorithms to track tangibles as small as 24×24mm with a high accuracy of 91.9% and a small rotation MAE of 3.85. We conditioned the network on AprilTag 36h11 markers and demonstrated that the network generalizes well to unseen AprilTag 16h5, ArUco original, and ARToolKit 4×4 markers. Furthermore, we deployed the network on a commodity tablet and achieved real-time fiducial tracking with 124ms inference duration. We performed a real-world user evaluation for tangible interaction showing average detection times between 832ms and 2231ms. We presented showcases that improve the interactive experience with tangibles and highlight the potential of super-resolved fiducial markers for security and authentication. The network, data, and code are publicly available via https://github.com/mimuc/super-resolution-for-fiducial-tangibles.

Despite their potential and the large body of research, commercially available tangible interfaces remain scarce. In the long term, we wish to bridge this gap and seamlessly integrate tangibles on commodity capacitive touchscreens. For this, we envision a collaboration with manufacturers and tech companies with the joint goal of improving the interactive experience on capacitive touchscreens.

[11] https://developer.qualcomm.com/qualcomm-robotics-rb5-kit/software-reference-manual/machine-learning/tensorflow

REFERENCES

[1] Daniel Avrahami, Jacob O. Wobbrock, and Shahram Izadi. 2011. Portico: Tangible Interaction on and around a Tablet. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 347–356. https://doi.org/10.1145/2047196.2047241

[2] Simon Baker and Takeo Kanade. 2002. Limits on Super-Resolution and How to Break Them. *IEEE Trans. Pattern Anal. Mach. Intell.* 24, 9 (2002), 1167–1183. https://doi.org/10.1109/TPAMI.2002.1033210

[3] Patrick Baudisch, Torsten Becker, and Frederik Rudeck. 2010. Lumino: Tangible Blocks for Tabletop Computers Based on Glass Fiber Bundles. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10)*. Association for Computing Machinery, New York, NY, USA, 1165–1174. https://doi.org/10.1145/1753326.1753500

[4] Mads Bock, Martin Fisker, Kasper Fischer Topp, and Martin Kraus. 2015. Tangible Widgets for a Multiplayer Tablet Game in Comparison to Finger Touch. In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play* (London, United Kingdom) *(CHI PLAY '15)*. Association for Computing Machinery, New York, NY, USA, 755–758. https://doi.org/10.1145/2793107.2810269

[5] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. 2016. Unsupervised Pixel-Level Domain Adaptation with Generative Adversarial Networks. https://doi.org/10.48550/ARXIV.1612.05424

[6] John Campbell and Xharmagne Carandang. 2012. Comparing Graphical and Tangible User Interfaces for a Tower Defense Game. In *18th Americas Conference on Information Systems (AMCIS 2012)*. Association for Information Systems, New York, NY, USA, 10. https://aisel.aisnet.org/amcis2012/proceedings/HCIStudies/11

[7] 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 SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2189–2192. https://doi.org/10.1145/2207676.2208371

[8] Leida Chen and Ravi Nath. 2008. A Socio-Technical Perspective of Mobile Work. *Inf. Knowl. Syst. Manag.* 7, 1,2 (2008), 41–60.

[9] 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. , 9 pages. https://doi.org/10.1007/978-3-030-00928-1_11

[10] Yu Cheng, Zhe Gan, Yitong Li, Jingjing Liu, and Jianfeng Gao. 2020. Sequential Attention GAN for Interactive Image Editing. In *Proceedings of the 28th ACM International Conference on Multimedia*. Association for Computing Machinery, New York, NY, USA, 4383–4391. https://doi.org/10.1145/3394171.3413551

[11] Frederick Choi, Sven Mayer, and Chris Harrison. 2021. 3D Hand Pose Estimation on Conventional Capacitive Touchscreens. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction*. Association for Computing Machinery, New York, NY, USA, Article 3, 13 pages. https://doi.org/10.1145/3447526.3472045

[12] Okkyung Choi, Seolhwa Han, Seungbin Moon, Kangseok Kim, Hongjin Yeh, and Taesik Shon. 2013. Secure Mobile Payment Service Using Vibration Cues on Near Field Communication Smartphone. *Sensor Letters* 11 (2013), 1750–1754. https://doi.org/10.1166/sl.2013.2994

[13] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A. Bharath. 2018. Generative Adversarial Networks: An Overview. *IEEE Signal Processing Magazine* 35, 1 (2018), 53–65. https://doi.org/10.1109/msp.2017.2765202

[14] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2014. Learning a Deep Convolutional Network for Image Super-Resolution. In *Computer Vision – ECCV 2014*. Springer International Publishing, Cham, 184–199.

[15] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2016. Image Super-Resolution Using Deep Convolutional Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 38, 2 (2016), 295–307. https://doi.org/10.1109/TPAMI.2015.2439281

[16] 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*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376772

[17] Shreyas Fadnavis. 2014. Image Interpolation Techniques in Digital Image Processing: An Overview. *International Journal Of Engineering Research and Application* 4 (2014), 2248–962270.

[18] Min Fan, Alissa N. Antle, Maureen Hoskyn, Carman Neustaedter, and Emily S. Cramer. 2017. Why Tangibility Matters: A Design Case Study of At-Risk Children Learning to Read and Spell. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1805–1816.

[19] Min Fan, Uddipana Baishya, Elgin-Skye Mclaren, Alissa N. Antle, Shubhra Sarker, and Amal Vincent. 2018. Block Talks: A Tangible and Augmented Reality Toolkit for Children to Learn Sentence Construction. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI EA '18)*. Association for Computing Machinery, New York, NY, USA, 1–6. https://doi.org/10.1145/3170427.3188576

[20] Sina Farsiu, Dirk Robinson, Michael Elad, and Peyman Milanfar. 2004. Advances and challenges in super-resolution. *International Journal of Imaging Systems and Technology* 14, 2 (2004), 47–57. https://doi.org/10.1002/ima.20007

[21] Mark Fiala. 2005. ARTag, a fiducial marker system using digital techniques. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2 (2005), 590–596 vol. 2. https://doi.org/10.1109/CVPR.2005.74

[22] George W. Fitzmaurice, Hiroshi Ishii, and William A. S. Buxton. 1995. Bricks: Laying the Foundations for Graspable User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '95)*. ACM Press/Addison-Wesley Publishing Co., USA, 442–449. https://doi.org/10.1145/223904.223964

[23] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel J. Marín-Jiménez. 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition* 47, 6 (2014), 2280–2292. https://doi.org/10.1016/j.patcog.2014.01.005

[24] Steven Gelineck, Jesper Andersen, and Morten Büchert. 2013. Music Mixing Surface. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces* (St. Andrews, Scotland, United Kingdom) *(ITS '13)*. Association for Computing Machinery, New York, NY, USA, 433–436. https://doi.org/10.1145/2512349.2517248

[25] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Networks. (2014). https://doi.org/10.48550/ARXIV.1406.2661

[26] 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* (Corfu, Island, Greece) *(PETRA '16)*. Association for Computing Machinery, New York, NY, USA, Article 22, 8 pages. https://doi.org/10.1145/2910674.2910690

[27] Timo Götzelmann and Daniel Schneider. 2016. CapCodes: Capacitive 3D Printable Identification and On-Screen Tracking for Tangible Interaction. In *Proceedings of the 9th Nordic Conference on Human-Computer Interaction* (Gothenburg, Sweden) *(NordiCHI '16)*. Association for Computing Machinery, New York, NY, USA, Article 32, 4 pages. https://doi.org/10.1145/2971485.2971518

[28] 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* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 3293–3315. https://doi.org/10.1145/3025453.3025808

[29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. https://doi.org/10.48550/ARXIV.1512.03385

[30] Michael S. Horn, Erin Treacy Solovey, and Robert J. K. Jacob. 2008. Tangible Programming and Informal Science Learning: Making TUIs Work for Museums. In *Proceedings of the 7th International Conference on Interaction Design and Children* (Chicago, Illinois) *(IDC '08)*. Association for Computing Machinery, New York, NY, USA, 194–201. https://doi.org/10.1145/1463689.1463756

[31] Meng-Ju Hsieh, Rong-Hao Liang, Jr-Ling Guo, and Bing-Yu Chen. 2018. RFIDesk: An Interactive Surface for Multi-Touch and Rich-ID Stackable Tangible Interactions. In *SIGGRAPH Asia 2018 Emerging Technologies* (Tokyo, Japan) *(SA '18)*. Association for Computing Machinery, New York, NY, USA, Article 11, 2 pages. https://doi.org/10.1145/3275476.3275491

[32] Yizhen Huang and Yangjing Long. 2006. Super-Resolution using Neural Networks Based on the Optimal Recovery Theory. In *16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*. Springer, Cham, Switzerland, 465–470. https://doi.org/10.1109/MLSP.2006.275595

[33] 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* (Singapore, Singapore) *(AH '15)*. Association for Computing Machinery, New York, NY, USA, 225–226. https://doi.org/10.1145/2735711.2735783

[34] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. 2016. Image-to-Image Translation with Conditional Adversarial Networks. https://doi.org/10.48550/ARXIV.1611.07004

[35] Tero Karras, Samuli Laine, and Timo Aila. 2018. A Style-Based Generator Architecture for Generative Adversarial Networks. https://doi.org/10.48550/ARXIV.1812.04948

[36] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2019. Analyzing and Improving the Image Quality of StyleGAN. https://doi.org/10.48550/ARXIV.1912.04958

[37] Sokratis Kartakis, Margherita Antona, and Constantine Stephanidis. 2011. Control Smart Homes Easily with Simple Touch. In *Proceedings of the 2011 International ACM Workshop on Ubiquitous Meta User Interfaces* (Scottsdale, Arizona, USA) *(Ubi-MUI '11)*. Association for Computing Machinery, New York, NY, USA, 1–6. https://doi.org/10.1145/2072652.2072654

[38] Hirokazu Kato and Mark Billinghurst. 1999. Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System. In *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR '99)*. IEEE Computer Society, New York, NY, USA, 85.

[39] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2015. Accurate Image Super-Resolution Using Very Deep Convolutional Networks. https://doi.org/10.48550/ARXIV.1511.04587

[40] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2015. Deeply-Recursive Convolutional Network for Image Super-Resolution. https://doi.org/10.48550/ARXIV.1511.04491

[41] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. https://doi.org/10.48550/ARXIV.1412.6980

[42] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. https://doi.org/10.48550/ARXIV.2010.05646

[43] Sven Kratz, Tilo Westermann, Michael Rohs, and Georg Essl. 2011. CapWidgets: Tangile Widgets versus Multi-Touch Controls on Mobile Devices. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI EA '11)*. Association for Computing Machinery, New York, NY, USA, 1351–1356. https://doi.org/10.1145/1979742.1979773

[44] Abinaya Kumar, Aishwarya Radjesh, Sven Mayer, and Huy Viet Le. 2019. Improving the Input Accuracy of Touchscreens Using Deep Learning. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI EA '19)*. Association for Computing Machinery, New York, NY, USA, 1–6. https://doi.org/10.1145/3290607.3312928

[45] Cornelius Lanczos. 1950. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. 45, 4 (1950).

[46] Kam Hon Lau, Umair Mujtaba Qureshi, Bruno Silva, and Gerhard Petrus Hancke. 2021. Mobile Proximity Channel Using Vibration. In *IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, New York, NY, USA, 1–6. https://doi.org/10.1109/IECON48115.2021.9589526

[47] 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* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173934

[48] 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)*. Association for Computing Machinery, New York, NY, USA, 779–792. https://doi.org/10.1145/3242587.3242605

[49] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, New York, NY, USA, 105–114. https://doi.org/10.1109/CVPR.2017.19

[50] Chuan Li and Michael Wand. 2016. Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks. https://doi.org/10.48550/

ARXIV.1604.04382

[51] Yanhong Li, Meng Liang, Julian Preissing, Nadine Bachl, Michelle Melina Dutoit, Thomas Weber, Sven Mayer, and Heinrich Hussmann. 2022. A Meta-Analysis of Tangible Learning Studies from the TEI Conference. In *Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Daejeon, Republic of Korea) *(TEI '22)*. Association for Computing Machinery, New York, NY, USA, Article 7, 17 pages. https://doi.org/10.1145/3490149.3501313

[52] Zhouchen Lin and Heung-Yeung Shum. 2004. Fundamental limits of reconstruction-based superresolution algorithms under local translation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 1 (2004), 83–97. https://doi.org/10.1109/TPAMI.2004.1261081

[53] Jen-Yu Liu, Yu-Hua Chen, Yin-Cheng Yeh, and Yi-Hsuan Yang. 2020. Unconditional Audio Generation with Generative Adversarial Networks and Cycle Regularization. In *Interspeech 2020*. ISCA, Baixas, French, 5. https://doi.org/10.21437/interspeech.2020-1137

[54] 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*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376189

[55] 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)*. Association for Computing Machinery, New York, NY, USA, 220–229. https://doi.org/10.1145/3132272.3134130

[56] Sven Mayer, Xiangyu Xu, and Chris Harrison. 2021. Super-Resolution Capacitive Touchscreens. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 12, 10 pages. https://doi.org/10.1145/3411764.3445703

[57] Allison Merz, Annie Hu, and Tracey Lin. 2018. ClipWorks: A Tangible Interface for Collaborative Video Editing. In *Proceedings of the 17th ACM Conference on Interaction Design and Children* (Trondheim, Norway) *(IDC '18)*. Association for Computing Machinery, New York, NY, USA, 497–500. https://doi.org/10.1145/3202185.3210758

[58] Mehdi Mirza and Simon Osindero. 2014. Conditional Generative Adversarial Nets. https://doi.org/10.48550/ARXIV.1411.1784

[59] Eric Daniel Mjolsness. 1986. *Neural Networks, Pattern Recognition, and Fingerprint Hallucination*. Ph. D. Dissertation. California Institute of Technology. https://doi.org/10.7907/M0VQ-DJ43

[60] Kamal Nasrollahi and Thomas B Moeslund. 2014. Super-resolution: a comprehensive survey. *Machine vision and applications* 25, 6 (2014), 1423–1468. https://doi.org/10.1007/s00138-014-0623-4

[61] Alexander Ng, Stephen A. Brewster, Frank Beruscha, and Wolfgang Krautter. 2017. An Evaluation of Input Controls for In-Car Interactions. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2845–2852. https://doi.org/10.1145/3025453.3025736

[62] Edwin Olson. 2011. AprilTag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*. IEEE, New York, NY, USA, 3400–3407. https://doi.org/10.1109/ICRA.2011.5979561

[63] Nobuyuki Otsu. 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* 9, 1 (1979), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

[64] Pankaj Parsania and Dr V.Virparia. 2015. A Review: Image Interpolation Techniques for Image Scaling. *International Journal of Innovative Research in Computer and Communication Engineering* 02 (2015), 7409–7414. https://doi.org/10.15680/IJIRCCE.2014.0212024

[65] Santiago Pascual, Antonio Bonafonte, and Joan Serrà. 2017. SEGAN: Speech Enhancement Generative Adversarial Network. https://doi.org/10.48550/ARXIV.1703.09452

[66] Esben Warming Pedersen and Kasper Hornbæk. 2011. Tangible Bots: Interaction with Active Tangibles in Tabletop Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 2975–2984. https://doi.org/10.1145/1978942.1979384

[67] Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. https://doi.org/10.48550/ARXIV.1511.06434

[68] Jun Rekimoto. 2002. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Minneapolis, Minnesota, USA) *(CHI '02)*. Association for Computing Machinery, New York, NY, USA, 113–120. https://doi.org/10.1145/503376.503397

[69] David E Rumelhart, Richard Durbin, Richard Golden, and Yves Chauvin. 1995. Backpropagation: The basic theory. *Backpropagation: Theory, architectures and applications* (1995), 1–34.

[70] Martin Schmitz, Florian Müller, Max Mühlhäuser, Jan Riemann, and Huy Viet Viet Le. 2021. Itsy-Bits: Fabrication and Recognition of 3D-Printed Tangibles with Small Footprints on Capacitive Touchscreens. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 419, 12 pages. https://doi.org/10.1145/3411764.3445502

[71] 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* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 1001–1014. https://doi.org/10.1145/3025453.3025663

[72] Wenzhe Shi, Jose Caballero, Ferenc Huszar, 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 (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 1874–1883. https://doi.org/10.1109/CVPR.2016.207 arXiv:1609.05158

[73] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. 2016. Learning from Simulated and Unsupervised Images through Adversarial Training. https://doi.org/10.48550/ARXIV.1612.07828

[74] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2014. Striving for Simplicity: The All Convolutional Net. https://doi.org/10.48550/ARXIV.1412.6806

[75] Benedict Steuerlein and Sven Mayer. 2022. Conductive Fiducial Tangibles for Everyone: A Data Simulation-Based Toolkit Using Deep Learning. *Proc. ACM Hum.-Comput. Interact.* 6, MHCI, Article 183 (sep 2022), 22 pages. https://doi.org/10.1145/3546718

[76] Paul Streli and Christian Holz. 2021. CapContact: Super-Resolution Contact Areas from Capacitive Touchscreens. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 289, 14 pages. https://doi.org/10.1145/3411764.3445621

[77] Satoshi Suzuki and KeiichiA be. 1985. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing* 30, 1 (1985), 32–46. https://doi.org/10.1016/0734-189X(85)90016-7

[78] R Tsai. 1984. Multiframe image restoration and registration. *Advance Computer Visual and Image Processing* 1 (1984), 317–339.

[79] Brygg Ullmer and Hiroshi Ishii. 1997. The MetaDESK: Models and Prototypes for Tangible User Interfaces. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology* (Banff, Alberta, Canada) *(UIST '97)*. Association for Computing Machinery, New York, NY, USA, 223–232. https://doi.org/10.1145/263407.263551

[80] John Underkoffler and Hiroshi Ishii. 1999. Urp: A Luminous-Tangible Workbench for Urban Planning and Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) *(CHI '99)*. Association for Computing Machinery, New York, NY, USA, 386–393. https://doi.org/10.1145/302979.303114

[81] 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 & Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 351–356. https://doi.org/10.1145/2807442.2807466

[82] 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* (St. Andrews, Scotland, United Kingdom) *(ITS '13)*. Association for Computing Machinery, New York, NY, USA, 101–104. https://doi.org/10.1145/2512349.2512791

[83] Daniel Wagner and Dieter Schmalstieg. 2007. ARToolKitPlus for Pose Tracking on Mobile Devices. In *Proceedings of 12th Computer Vision Winter Workshop (CVWW'07)*. Graz Technical University, St. Lambrecht, Austria, 8.

[84] Johnty Wang, Nicolas D'Alessandro, Sidney Fels, and Bob Pritchard. 2011. SQUEEZY: Extending a Multi-touch Screen with Force Sensing Objects for Controlling Articulatory Synthesis. In *11th International Conference on New Interfaces for Musical Expression (NIME 2011)*. nime.org, 531–532.

[85] John Wang and Edwin Olson. 2016. AprilTag 2: Efficient and robust fiducial detection. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, New York, NY, USA, 4193–4198. https://doi.org/10.1109/IROS.2016.7759617

[86] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, and Xiaoou Tang. 2018. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. https://doi.org/10.48550/ARXIV.1809.00219

[87] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612. https://doi.org/10.1109/TIP.2003.819861

[88] 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 SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 481–490. https://doi.org/10.1145/1518701.1518779

[89] Bartlomiej Wronski, Ignacio Garcia-Dorado, Manfred Ernst, Damien Kelly, Michael Krainin, Chia-Kai Liang, Marc Levoy, and Peyman Milanfar. 2019. Handheld Multi-Frame Super-Resolution. *ACM Trans. Graph.* 38, 4, Article 28 (2019), 18 pages. https://doi.org/10.1145/3306346.3323024

[90] Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T. Freeman, and Joshua B. Tenenbaum. 2016. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. https://doi.org/10.48550/ARXIV.1610.07584

[91] Robert Xiao, Sven Mayer, and Chris Harrison. 2020. VibroComm: Using Commodity Gyroscopes for Vibroacoustic Data Reception. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services* (Oldenburg, Germany) *(MobileHCI '20)*. Association for Computing Machinery, New York, NY, USA, Article 5, 9 pages. https://doi.org/10.1145/3379503.3403540

[92] 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)*. Association for Computing Machinery, New York, NY, USA, 47–50. https://doi.org/10.1145/2817721.2817737

[93] Xiangyu Xu, Yongrui Ma, and Wenxiu Sun. 2019. Towards Real Scene Super-Resolution with Raw Images. https://doi.org/10.48550/ARXIV.1905.12156

[94] Chih-Yuan Yang, Chao Ma, and Ming-Hsuan Yang. 2014. Single-Image Super-Resolution: A Benchmark. In *Computer Vision – ECCV 2014*. Springer International Publishing, Cham, Switzerland, 372–386. https://doi.org/10.1007/978-3-319-10593-2_25

[95] Xin Yu and Fatih Porikli. 2016. Ultra-Resolving Face Images by Discriminative Generative Networks. In *Computer Vision – ECCV 2016*. Springer International Publishing, Cham, Switzerland, 318–333. https://doi.org/10.1007/978-3-319-46454-1_20

[96] Guillaume Zufferey, Patrick Jermann, Aurélien Lucchi, and Pierre Dillenbourg. 2009. TinkerSheets: Using Paper Forms to Control and Visualize Tangible Simulations. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction* (Cambridge, United Kingdom) *(TEI '09)*. Association for Computing Machinery, New York, NY, USA, 377–384. https://doi.org/10.1145/1517664.1517740