Physical Sketching Tools and Techniques for Customized Sensate Surfaces

UNIVERSITÄT DES SAARLANDES

A dissertation submitted towards the degree

Doctor of Engineering (Dr.-Ing.)

of the Faculty of Mathematics and Computer Science
of Saarland University

by

Narjes Pourjafarian

Saarbrücken, 2023

Date of the colloquium: 7th June 2023

Dean of the Faculty: Prof. Dr. Jürgen Steimle

Reviewers: Prof. Dr. Jürgen Steimle

Prof. Dr. Susanne Boll

Chair of the committee: Prof. Dr. Eddy Ilg

Academic assistant: Dr. Alice Haynes

Abstract

Sensate surfaces are a promising avenue for enhancing human interaction with digital systems due to their inherent intuitiveness and natural user interface. Recent technological advancements have enabled sensate surfaces to surpass the constraints of conventional touchscreens by integrating them into everyday objects, creating interactive interfaces that can detect various inputs such as touch, pressure, and gestures. This allows for more natural and intuitive control of digital systems. However, prototyping interactive surfaces that are customized to users' requirements using conventional techniques remains technically challenging due to limitations in accommodating complex geometric shapes and varying sizes. Furthermore, it is crucial to consider the context in which customized surfaces are utilized, as relocating them to fabrication labs may lead to the loss of their original design context. Additionally, prototyping high-resolution sensate surfaces presents challenges due to the complex signal processing requirements involved. This thesis investigates the design and fabrication of customized sensate surfaces that meet the diverse requirements of different users and contexts. The research aims to develop novel tools and techniques that overcome the technical limitations of current methods and enable the creation of sensate surfaces that enhance human interaction with digital systems.

Zusammenfassung

Sensorische Oberflächen sind aufgrund ihrer inhärenten Intuitivität und natürlichen Benutzeroberfläche ein vielversprechender Ansatz, um die menschliche Interaktion mit digitalen Systemen zu verbessern. Die jüngsten technologischen Fortschritte haben es ermöglicht, dass sensorische Oberflächen die Beschränkungen herkömmlicher Touchscreens überwinden, indem sie in Alltagsgegenstände integriert werden und interaktive Schnittstellen schaffen, die diverse Eingaben wie Berührung, Druck, oder Gesten erkennen können. Dies ermöglicht eine natürlichere und intuitivere Steuerung von digitalen Systemen. Das Prototyping interaktiver Oberflächen, die mit herkömmlichen Techniken an die Bedürfnisse der Nutzer angepasst werden, bleibt jedoch eine technische Herausforderung, da komplexe geometrische Formen und variierende Größen nur begrenzt berücksichtigt werden können.

Darüber hinaus ist es von entscheidender Bedeutung, den Kontext, in dem diese individuell angepassten Oberflächen verwendet werden, zu berücksichtigen, da eine Verlagerung in Fabrikations-Laboratorien zum Verlust ihres ursprünglichen Designkontextes führen kann. Zudem stellt das Prototyping hochauflösender sensorischer Oberflächen aufgrund der komplexen Anforderungen an die Signalverarbeitung eine Herausforderung dar. Diese Arbeit erforscht das Design und die Fabrikation individuell angepasster sensorischer Oberflächen, die den diversen Anforderungen unterschiedlicher Nutzer und Kontexte gerecht werden. Die Forschung zielt darauf ab, neuartige Werkzeuge und Techniken zu entwickeln, die die technischen Beschränkungen derzeitiger Methoden überwinden und die Erstellung von sensorischen Oberflächen ermöglichen, die die menschliche Interaktion mit digitalen Systemen verbessern.

Acknowledgements

I am deeply grateful to all those who have supported and encouraged me throughout this challenging yet rewarding journey. To begin with, I would like to express my heartfelt thanks to my advisor, Prof. Jürgen Steimle, for his invaluable guidance, unwavering support, and numerous thought-provoking discussions. His mentorship and dedication to my academic growth have been crucial in shaping my research and career.

I also want to extend my appreciation to my colleagues and friends for creating a stimulating and collaborative environment. I am particularly grateful to Dr. Marion Koelle and Dr. Paul Strohmeier for their continuous support and constructive feedback throughout my PhD.

Furthermore, I am grateful to Dr. Aditya Shekhar Nittala, Dr. Adwait Sharma, Dr. Anusha Withana, Dr. Arshad Khan, and Dr. Bruno Fruchard for their constant inspiration. I have also benefitted from the helpful discussions with Dr. Alice Haynes, Dr. Martin Schmitz, and Dr. Ata Otaran. I am grateful for the friendship of Artin Saberpour, Marie Muehlhaus, Madalina Nicolae, and Arata Jingu. Additionally, I had the pleasure of working with many talented students, in particular Fjolla Mjaku, Sahar Mavali, and Sina Mavali. Their hard work was a tremendous help to my research.

Last but not least, I want to express my deepest appreciation to my family and friends for their unconditional love, encouragement, and support throughout this journey. Their presence in my life has been a source of strength and motivation, and I am forever grateful for their unwavering support.

Contents

1 Introduction

Sensate surfaces have emerged as a highly promising area of research, offering the potential to transform the way humans interact with digital systems. These surfaces are designed to detect various forms of human input, such as touch, pressure, or gestures [58; 226; 242], creating highly intuitive and natural user interfaces [188; 259]. With the advancements in technology, sensate surfaces have transcended traditional touchscreens and are now integrated into everyday objects such as clothing [161], furniture [240], and even on the human body [237; 239], creating more natural and intuitive interactions. For instance, the use of sensate surfaces on the skin has enabled discreet and hands-free control of electronic devices, such as adjusting the volume of a device or controlling music playback with simple hand gestures. Furthermore, incorporating these surfaces into furniture has facilitated the seamless integration of technology into living spaces, such as detecting when someone is sitting down and automatically adjusting the lighting or temperature to create a more comfortable environment. Given the versatile and efficient nature of sensate surfaces, with the ability to detect diverse forms of input or provide haptic feedback, they have significant implications for a wide range of applications, including smart homes, education, and entertainment.

Recent advances in digital fabrication have brought about a paradigm shift in the way we create interfaces. This has opened up new avenues for creating highly customized interactive surfaces tailored to the specific needs of an individual or environment. The ability to personalize interfaces has far-reaching implications for user experience, ergonomics, and design flexibility. The level of personalization provided by these interfaces creates a more intuitive and natural way of interaction, where the interface is adapted to the user's individual needs and preferences. For instance, the physical form and layout of interfaces can be tailored to fit the user's physical needs and preferences, such as size, shape, and placement [96; 121; 237]. Moreover, the design flexibility that digital fabrication techniques offer enables the creation of unique and distinctive interfaces that can be aesthetically pleasing and functional [60; 240]. This high level of personalization allows for the development of new types of sensors, including highly customized multi-touch sensors on everyday objects that support more immersive forms of interaction.

Prototyping plays a crucial role in the design of such customized interactive systems [16]. This practice offers designers and makers the chance to experiment with different materials and features, thereby facilitating the exploration and improvement of ideas before finalizing a design [21; 192]. The rapid and iterative nature of prototyping accelerates the creative process, thus enabling designers to realize their concepts more quickly. Despite its significance, prototyping customized interactive surfaces with embedded functionality through conventional techniques remains a technically challenging task. Conventional fabrication techniques may not accommodate objects with complex geometric shapes and varying sizes. For instance, a 3D object cannot fit inside a conventional 2D inkjet printer, and it is not feasible to move an immobile object to a screen printing lab. Moreover, the context in which these objects are utilized is also critical, and relocating them may result in the loss of the original design context. Fabricating interfaces on objects with rich affordances, such as soft and deformable objects, presents further challenges due to the tension and strain they cause during the fabrication process, which necessitates sophisticated machinery and a specialized fabrication process.

In addition, creating customized interfaces using existing methods separates the digital design and fabrication steps from the intended surface, neglecting the intended use and inherent features of the objects, such as their shape, size, texture, and material. The prototyping of high-resolution sensate surfaces also poses challenges due to the complex signal processing requirements involved. These limitations hinder the development of interactive surfaces that can integrate seamlessly into the user's personal environment or conform to the shape of personal objects or the human body, thus impacting the range of interactive devices available for wearable and ubiquitous computing applications.

Therefore, designing and prototyping customized interactive surfaces require tailored and innovative approaches that consider the context and complex geometries of each object. These approaches must be able to accommodate the creation of interfaces on soft and deformable objects to meet their unique requirements. Additionally, the digital design and fabrication steps must be integrated to allow designers to develop and test their ideas in a relevant context. Furthermore, facilitating the creation and read-out of high-resolution sensate surfaces is essential to enhance interaction. The aim of this thesis is to combine the expressiveness and ease of physical sketching with the support of digital fabrication tools to address these challenges. Physical sketching tools and techniques will be investigated to prototype customized sensate surfaces alongside DIY sensing techniques to facilitate signal processing on high-resolution interfaces. These approaches consider the distinct attributes of sensate surfaces and provide a practical solution for creating interactive devices that can be customized to meet individual user needs.

1.1 Physical Sketching of Functional Interfaces

Physical sketching is a key activity in the prototyping stage and serves as a low-cost, low-fidelity method for evaluating and communicating design concepts. As a fundamental aspect of the design process, sketching provides designers with a quick and efficient way to visualize their ideas. The iterative and incremental nature of sketching allows for expedited exploration of various design concepts, including the examination of different shapes, textures, and proportions of surfaces [31]. Given its ease and simplicity, physical sketching is considered a highly valuable tool, even in circuit design.

Physical Sketching Techniques

Sketching in circuits is a rapid prototyping technique that employs low-fidelity materials such as conductive ink or copper tape for the quick and iterative design exploration of electronic circuits [125; 170]. This method presents a promising approach for creating customized interactive surfaces that can be integrated into our surrounding environment. It offers an intuitive and efficient way of creating tangible interfaces and functional prototypes on complex geometries and rich materials. Moreover, creating the circuit sketch directly on the desired object allows us to immediately experience how the affordances of the object and the physical environment impact the interaction.

The process of prototyping interactive surfaces has been explored using various methods, ranging from manual sketching to digital fabrication techniques. The manual approach involves the use of a conductive stylus or conductive paint to sketch circuits and sensors directly onto the surface. For example, Buechley et al. [17] employed conductive paint on paper as a wire to connect computational elements to each other. Mellis et al. [125] extended this approach by developing a simple and robust technique for drawing circuits with conductive ink on paper, thereby facilitating the direct integration of off-the-shelf electronic components into interactive artifacts. This approach enables users to engage in incremental and iterative sketching, which is crucial in the ideation phase. However, it is limited in terms of fabrication speed and precision. Digital fabrication techniques, on the other hand, offer greater precision and speed in producing intricate designs, repetitive patterns, and fine details. For instance, Kawahara et al. [98] proposed a method for printing functional circuits using a desktop inkjet printer. Khan et al. [99] extended this approach by developing a method for printing soft circuits using an inkjet printer. Choi et al. [28] presented a handheld 2D plotter that allows users to print circuits and sensors directly onto the skin. However, such techniques often require less direct user engagement in the fabrication process, which can hinder creativity.

This thesis aims to investigate the fabrication techniques that fall between these two extremes, exploring the potential of using physical sketching tools that enable designers to focus on their creativity while delegating tasks to machines. This approach allows for improved accuracy in sketches while fostering creative engagement between the designer and the fabrication process. By exploring the use of physical sketching tools, this thesis aims to contribute to the development of more effective and efficient fabrication techniques that empower designers to create more sophisticated and innovative designs. The potential implications of such an approach could have far-reaching consequences for interactive surfaces, enabling greater flexibility and creativity in the design and fabrication of circuits, sensors, and aesthetic elements.

Touch Sensing Technique

Sketching in hardware refers to the process of rapidly and iteratively creating physical prototypes of electronic devices such as circuits and sensors. It is an approach that aims to provide solutions for designers, makers, and laypeople to easily realize complete and functional hardware prototypes, involving not only the fabrication of interfaces but also the entire pipeline, including signal readout and processing, and integration into a functional application. The Arduino platform is widely popular, providing accessible means of creating functional interfaces. However, despite existing solutions for creating input and output interfaces using Arduino, the recognition of multi-touch input remains a major challenge yet to be resolved.

Multi-touch sensing is an important input modality that enables more natural and intuitive forms of user interaction as well as recognition of complex gestures and inputs, making it an essential part of many interactive experiences. However, due to the complexity of electrical and signal processing requirements, it remains challenging to create interface prototypes with custom-designed multi-touch input surfaces using Arduino or other DIY techniques. In this thesis, we investigate a DIY technique for high-resolution multi-touch sensing on sensate surfaces using a commodity microcontroller.

Consequently, the primary objective of this thesis is to address the major challenges encountered in developing physical sketching tools and techniques for creating sensate surfaces. To achieve this aim, this thesis will focus on the following specific goals:

1. The integration of the ease and flexibility of physical sketching with the precision of digital design methods to establish seamless and efficient fabrication techniques. Additionally, we will seek to develop accessible fabrication tools and artifacts that support physical sketching for the creation of sensate surfaces while considering electronic constraints and design aesthetics.
2. The examination of the requirements for physical sketching on a diverse range of objects, including rich materials, deformable and soft surfaces, and complex geometries such as skin. This will be done to ensure the preservation of form factor and visual-haptic properties.
3. The development of multi-touch input recognition and sensor read-out on high-resolution sensate surfaces for designers and makers without having electronic knowledge.

Through these goals, this thesis seeks to advance the field of physical sketching for the creation of sensate surfaces, facilitating the creation of more immersive and interactive user experiences.

1.2 Contributions of this Thesis

This thesis contributes to the field of interactive fabrication and rapid prototyping by introducing three novel physical sketching techniques for creating sensate surfaces and a DIY touch sensing approach (see Figure 1.1). The physical sketching tools and techniques lie between manual sketching and digital fabrication, providing new ways to advance the creation of interactive surfaces and allowing designers and makers to quickly and easily bring their ideas to life. The touch sensing technique allows for high-resolution multi-touch sensing on a diverse range of fabricated sensors, from manually sketched to digitally printed sensors. By providing new tools and techniques for hands-on prototyping and fabrication on complex geometries and rich materials, this thesis furthers the development of the field and opens up new avenues for research and exploration.

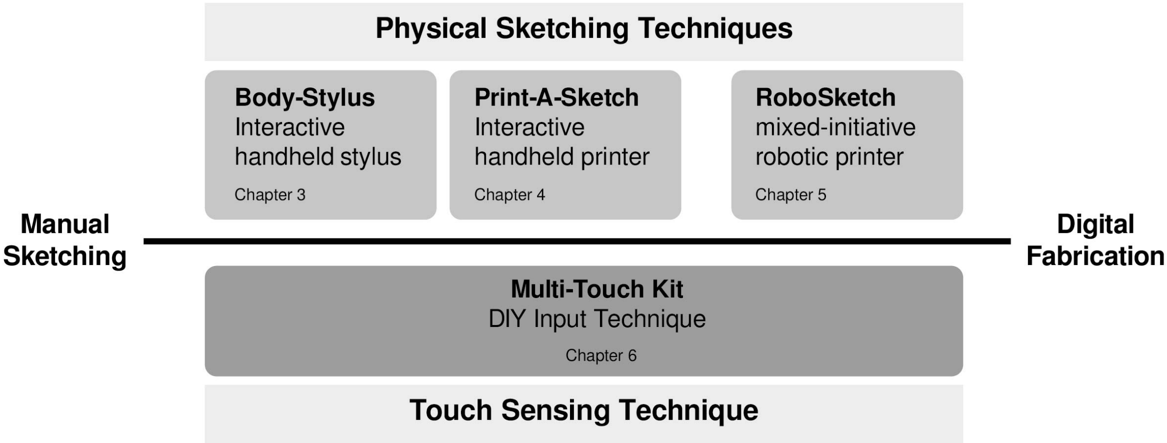


Figure 1.1 Physical sketching tools and techniques contributed to this thesis.

1.2.1 BodyStylus: Freehand On-body Fabrication

BodyStylus is our first contributed physical sketching technique for the in-situ design and fabrication of interactive interfaces on soft, deformable, and complex geometries such as the human body (Figure 1.2a). Inspired by traditional techniques of sketching on the skin, including tattooing, makeup, and henna art, we provide a physical sketching handheld tool that complements freehand inking with digital support. Projected in-situ guidance facilitates the creation of valid circuits, sensors, and aesthetic ornaments on the body that align with the human body landscape. The proactive switching between inking and non-inking modes also serves as a constraint mechanism to prevent errors during the design process. The design principles of *BodyStylus* are derived from the unique combination of aesthetics, the human body landscape, and circuit logic. A set of design techniques allows users to sketch visually aesthetic epidermal devices on the body quickly and directly. To demonstrate the practical feasibility of this approach, we present an interactive prototype that uses a stylus refilled with self-sintering conductive ink.

1.2.2 Print-A-Sketch: Handheld Fabrication on Rich Materials

The development of our first sketching tool, in the form of a stylus, was motivated by the need to create interfaces on complex geometries, such as the human body. However, to broaden the scope of the sketching canvas and include everyday surfaces with varied materials, it was necessary to consider the material properties and context of objects in the environment. With this in mind, a second physical sketching technique was devised

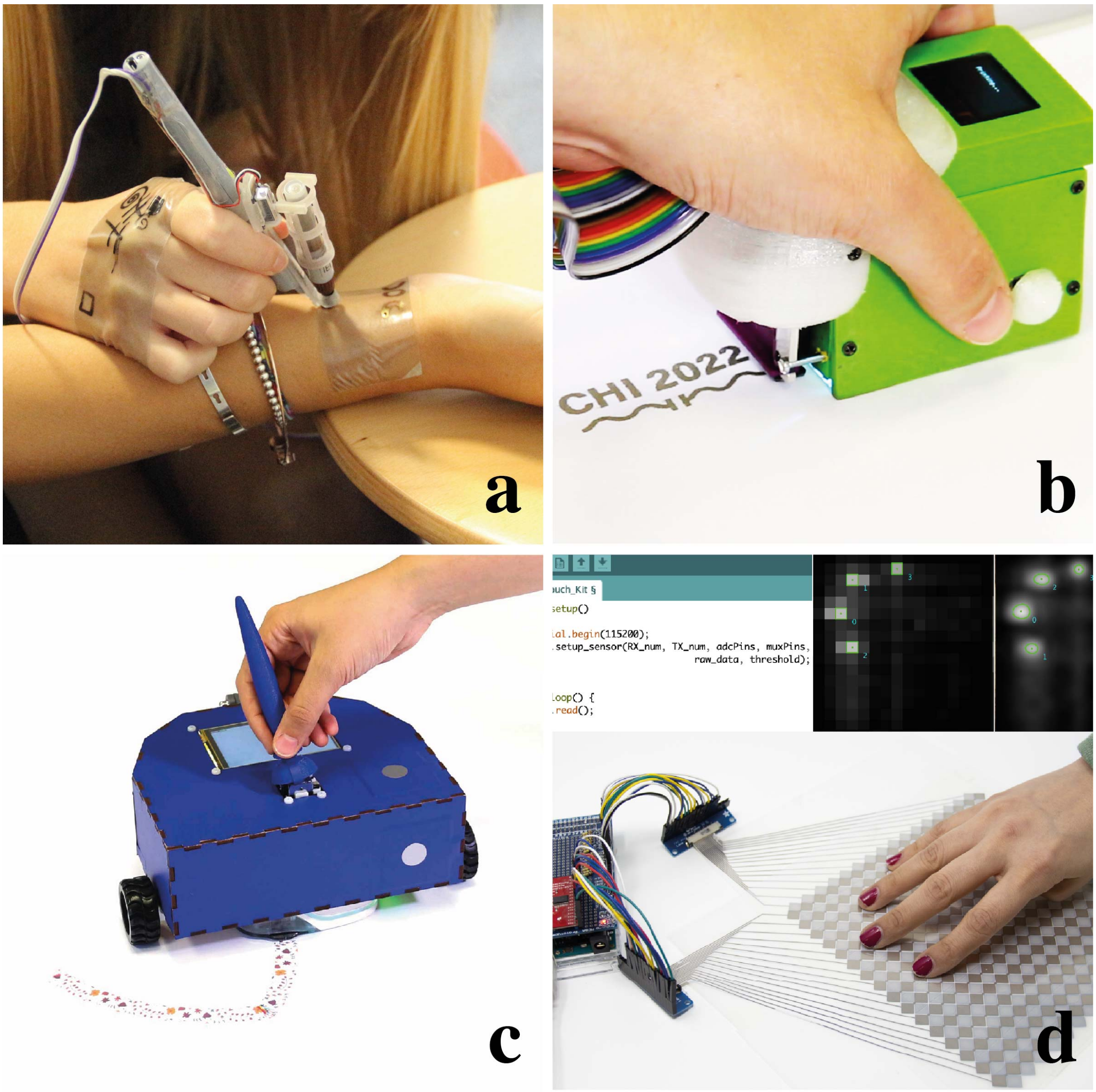


Figure 1.2 Physical sketching tools and techniques contributed to this thesis: a) BodyStylus, b) Print-A-Sketch, c) RoboSketch, d) Multi-touch Kit.

with the aim of enhancing precision, consistency, and the capability to print on everyday surfaces with diverse materials.

The resultant artifact, *Print-A-Sketch*, is a context-aware handheld printer designed for physical sketching circuits and sensors on everyday surfaces (Figure 1.2b). It brings together the desirable properties of freehand sketching and functional electronic printing, allowing for both manual controls of large strokes and computer control of fine details.

The shared control of *Print-A-Sketch* enables the sketching of interactive interfaces on everyday objects, including those with challenging materials or sizes for printing.

In Chapter 4, we provide an overview of the challenges involved in designing such a system and present how they can be addressed using context-aware, dynamic printing. Continuous sensing is utilized to ensure quality prints by adjusting the inking rate based on hand movement and material properties. This also allows for the print to adapt to previously printed traces, supporting incremental and iterative sketching. Our demonstration of example applications highlights the good conductivity on various materials and high spatial precision achieved by *Print-A-Sketch*.

1.2.3 RoboSketch: Mixed-Initiative Physical Sketching

The techniques above have provided powerful handheld tools for physical sketching on everyday surfaces and the human body. However, these devices are limited by the reach of the user's hand and arm movement. To extend the range of sketching and increase the speed of fabrication, we contributed our third sketching technique, *RoboSketch*.

With *RoboSketch*, we introduce a new class of physical sketching devices that combines the desirable features of a handheld tool and an autonomous fabrication robot (Figure 1.2c). This enables a seamless transition from manual and assisted to autonomous fabrication. We contribute the concept of mixed-initiative physical sketching, utilizing a working robotic printer that can be handheld for freehand sketching, provide interactive assistance during sketching, or move independently for computer-generated sketching. Additionally, we present interaction techniques that facilitate seamless transitions between modes, as well as sketching techniques that benefit from these transitions, such as extending or revisiting sketches. Chapter 5 presents results from a case study with seven sketchers, demonstrating that mixed-initiative physical sketching enhances the flexibility of computer-supported sketching.

1.2.4 Multi-Touch Kit: DIY Input Technique for Sensate Surfaces

The deployment of high-resolution multi-touch sensors presents an opportunity to enhance the interaction on everyday surfaces. Despite the availability of physical sketching techniques for the fabrication of touch sensors on various surfaces, the creation of functional interfaces necessitates the transfer of signals between the sensors and the corresponding hardware. However, commercial touch controllers are not designed to support customized sensors. To address this issue, we contribute a touch sensing technique, *Multi-Touch Kit*, enabling electronics novices to rapidly prototype customized capacitive multi-touch sensors (Figure 1.2d). In contrast to existing approaches, the proposed technique utilizes a commodity microcontroller and open-source software and does not require any specialized hardware. Chapter 6 presents the results of technical principles, implementation, and evaluations, demonstrating that our approach enables multi-touch sensors with a high spatial and temporal resolution, capable of accurately detecting multiple simultaneous touches. Additionally, Chapter 6 presents five application examples to illustrate the versatile uses of our approach for sensors of different scales, curvatures, and materials.

1.3 Publications

The ideas and figures of this thesis have been published as four full papers at the ACM Conferences on Human Factors in Computing Systems (CHI) [P1, P2, P3] and User Interface Software and Technology (UIST) [P4].

P1. Narjes Pourjafarian, Marion Koelle, Bruno Fruchard, Sahar Mavali, Konstantin Klamka, Daniel Groeger, Paul Strohmeier, and Jürgen Steimle. 2021. *BodyStylus: Freehand On-Body Design and Fabrication of Epidermal Interfaces.* In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21).

P2. Narjes Pourjafarian, Marion Koelle, Fjolla Mjaku, Paul Strohmeier, and Jürgen Steimle. 2022. *Print-A-Sketch: A Handheld Printer for Physical Sketching of Circuits and Sensors on Everyday Surfaces.* In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22).

P3. Narjes Pourjafarian, Fjolla Mjaku, Marion Koelle, Martin Schmitz, Jan Borchers and Jürgen Steimle. 2023. *RoboSketch: Mixed-Initiative Physical Sketching with a Robotic Printer.* In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23).

P4. Narjes Pourjafarian, Anusha Withana, Joseph A. Paradiso, and Jürgen Steimle. 2019. *Multi-Touch Kit: A Do-It-Yourself Technique for Capacitive Multi-Touch Sensing Using a Commodity Microcontroller.* In Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19).

In addition to the main publications, the author contributed to the following relevant publications.

P5. Nihar Sabnis, Dennis Wittchen, Courtney N. Reed, Narjes Pourjafarian, Jürgen Steimle, and Paul Strohmeier. 2023. *Haptic Servos: Self-Contained Vibrotactile*

Rendering System for Creating or Augmenting Material Experiences. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23).

P6. Paul Strohmeier, Narjes Pourjafarian, Marion Koelle, Cedric Honnet, Bruno Fruchard, and Jürgen Steimle. 2020. *Sketching On-Body Interactions Using PiezoResistive Kinesiology Tape.* In Proceedings of the Augmented Humans International Conference (AHs '20).

1.4 Structure of this Thesis

This thesis is structured into seven chapters, as follows:

- **Chapter 2** provides an overview of the current state of the art in the fields of interactive fabrication, sketching interfaces, electronic circuit fabrication, and touch sensing.
- **Chapter 3** presents *BodyStylus*, a handheld tool that augments freehand inking with digital support for on-body design and fabrication of epidermal interfaces.
- **Chapter 4** presents *Print-A-Sketch*, a handheld printer for the physical sketching of circuits and sensors on everyday surfaces that combines desirable properties from freehand sketching and functional electronic printing.
- **Chapter 5** presents *RoboSketch*, a mixed-initiative physical sketching robotic printer that can be handheld for freehand sketching, can provide interactive assistance during sketching, or move about for computer-generated sketches.
- **Chapter 6** presents *Multi-Touch Kit*, a DIY technique that enables high-resolution multi-touch sensing on surfaces with different sizes, geometries, and materials using an Arduino microcontroller.
- **Chapter 7** provides an overview of the key conclusions drawn in this thesis and suggests potential areas for future research.

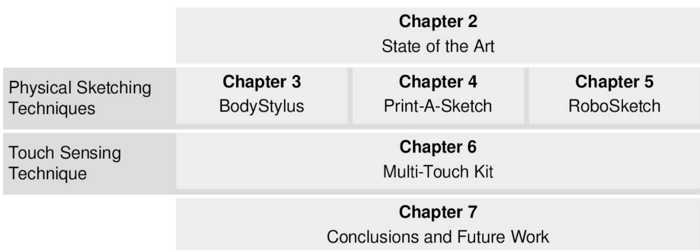


Figure 1.3 Structure of this thesis

2 Related Work

Physical sketching for fabrication of interactive surfaces is an emerging and interdisciplinary area of research that combines advances in interactive fabrication, sketching interfaces, fabricating electronic circuits, and touch sensing. The research in interactive fabrication focuses on the use of computer-controlled fabrication techniques with the included ability for real-time user interaction. Sketching interfaces, on the other hand, explore physical and digital tools and techniques for sketching and prototyping designs. The field of fabricating electronic circuits involves the design and fabrication of customized electronic circuits for use in interactive surfaces. Finally, touch sensing research examines the technologies for detecting touch input on sensate surfaces. This chapter aims to provide an overview of the current state of the art in these four areas, highlighting key research contributions and setting the stage for the research presented in this thesis.

2.1 Interactive Fabrication

Digital design and fabrication technology have revolutionized the way we create and customize objects. Utilizing computer-aided design (CAD) software, the design process can now be entirely digital, while fabrication is completed through the use of computer-controlled machines, such as 3D printers, inkjet printers, and laser cutters. This improves both the speed and accuracy of the fabrication process. However, creative activities often require user engagement *during* the fabrication process [7; 101]. Interactive fabrication, inspired by traditional crafting tools, allows for human participation throughout both the design and fabrication process, enabling real-time manipulation of the fabricated workpiece [243] (see Figure 2.1).

One example is Constructables [131], an interactive drafting table that allows users to draft directly on the workpiece using a handheld laser pointer, providing users with directness and precision in the fabrication process. The system tracks the pointer, and the high-powered laser cutter is used for visual feedback, rather than a screen or projection, resulting in the creation of simple but functional devices that cannot be made with traditional interactive fabrication tools. Another example is RoMA [154], an interactive fabrication system that uses a 3D printing robotic arm in conjunction with an AR CAD editor to provide designers with fast, precise, hands-on, and in-situ modeling experience. The system allows for quick interruption of printing to access a printed area and provides a tangible reference for the designer to add new elements to the design, allowing for the rapid integration of real-world constraints and the creation of well-proportioned tangible artifacts.

Additionally, FormFab [132] allows users to manipulate a thermoplastic sheet using a robotic arm and a pneumatic system to apply pressure or vacuum. Rather than adding or subtracting material, FormFab reshapes the material continuously, allowing users to interactively explore different shapes and sizes with a single interaction. More recently, Adroid [215] enables users to enhance precision and accuracy when using hand-held tools by utilizing a robotic arm as a virtual "jig", which constrains the tool's motion while preserving the user's autonomy in open-ended fabrication tasks. Further research has examined the use of interactive fabrication for a range of fabrication activities, including the creation of 3D models [155], fabricating e-textiles [105], directly controlling fabrication machines [52; 118; 210], and creating interfaces around the body [55; 56].

A comprehensive examination of all research on interactive fabrication at a large scale is beyond the scope of this thesis. The focus of this study is on interactive fabrication technologies for the creation of sensate surfaces. Therefore, this section examines interactive fabrication technologies that are used in situ and in context, those that are applied to the body, and interactive handheld tools. To provide a foundation for discussion, this section will first present background information on interactive fabrication in context.

2.1.1 Fabrication in Context

There are a variety of objects and surfaces that pose challenges in fabrication, such as large (e.g., furniture) or immobile objects (e.g., windows or doors) that are difficult to transport. These objects also possess significant contextual meaning (e.g., because of their orientation or location with respect to the environment), which is often lost when

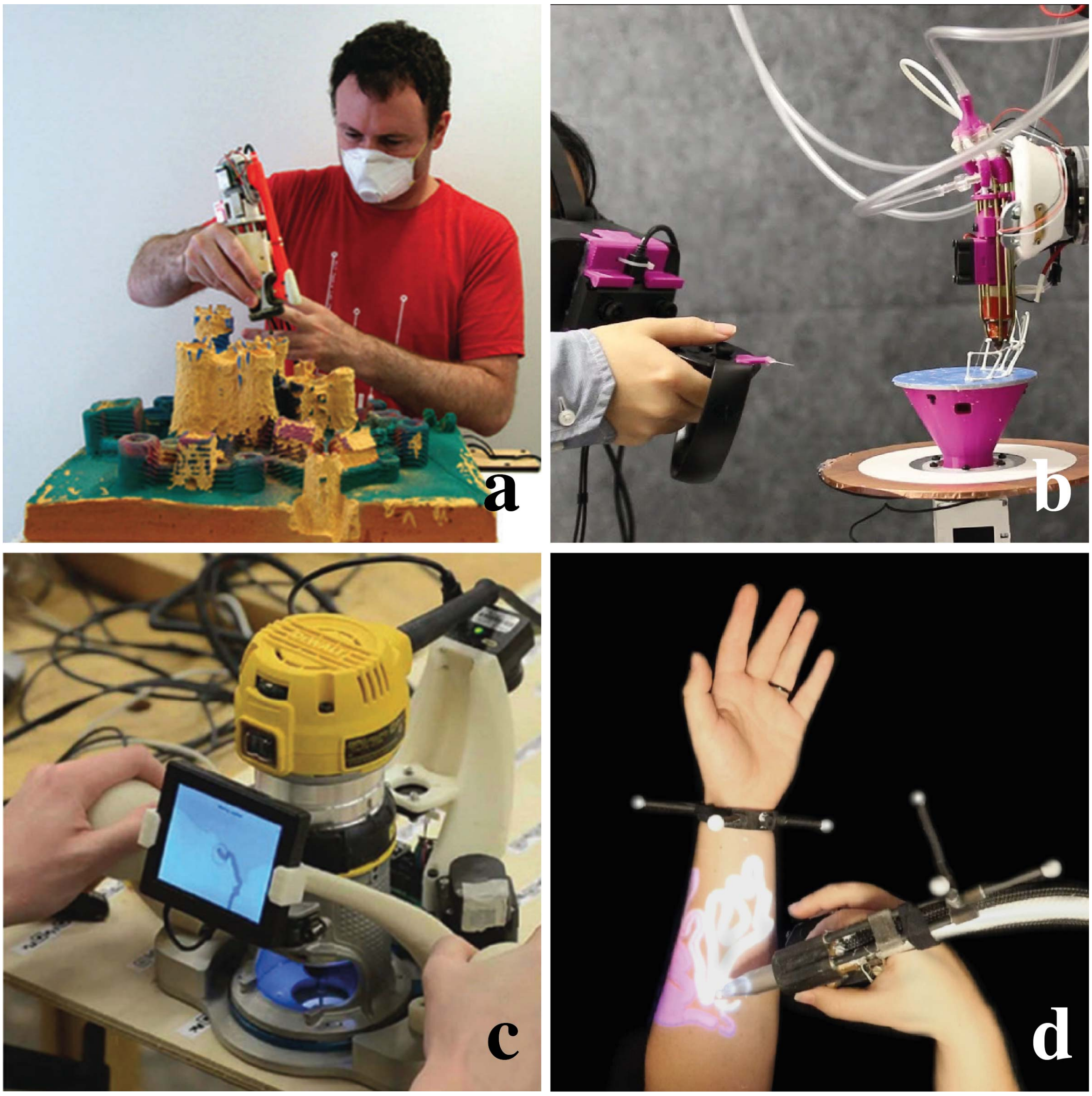


Figure 2.1 Examples of smart handheld tools for interactive fabrication include: a) FreeD, a freehand digital sculpting tool [265], b) RoMA, a robotic 3D printing assistant [154], c) ShaperTool, a position-correcting tool for precise large-scale cutting [179], and d) ExoSkin, a hybrid tool for printing digital artifacts directly on the body [56].

they are moved to a different location. As a result, it is desirable to be able to fabricate directly on objects and in place without the need of moving them or transporting them out of their respective context.

To address these challenges, in-place fabrication techniques have been developed. For example, Wessely et al. [240] have created sprayable interfaces which provide great flexibility in terms of geometry and scale of deployment. Sprayable User Interfaces present a novel approach to creating large-scale interactive surfaces by airbrushing functional inks, which allows designers to create user interfaces on complex 3D geometries where existing stationary fabrication methods fail. MixFab [235], a mixed-reality environment for personal fabrication allows users to design objects in an immersive augmented-reality environment and lowers the barrier for users to engage in personal fabrication, enabling them to introduce existing physical objects effortlessly into their designs. Additionally, Stemasov et al. [206] have introduced a mixed-reality system that allows users to browse model repositories, preview the models in-situ, and adapt them to their environment, with the aim of providing almost modeling-free personal fabrication for both novice and expert makers; and Roumen et al. [184] explored the concept of mobile personal fabrication, which refers to the ability to fabricate physical objects on the go using portable and handheld devices.

More recently, Interactive Robotic Plastering (IRoP) [129] has demonstrated a system that combines interactive design tools, an augmented reality interface, and a robotic spraying system, allowing designers and skilled workers to intuitively engage with an in-situ robotic plastering process; and sPrintr [22] has presented a mobile 3D printer that enables in-situ personal fabrication. The system consists of a compact 3D printer mounted on a robotic arm, which can be deployed in various locations to create custom objects. This type of fabrication is also common in eTextile research, as many textile craft methods are inherently in-place [66; 79; 91]. While in-place fabrication holds significant potential in enabling the preservation of contextual meaning in objects and surfaces, there are still many challenges that need to be addressed, such as in-place fabrication of high-resolution circuits and sensors on rich materials or around the body.

2.1.2 Fabrication around the Body

Interactive fabrication around the body has the potential to create objects and surfaces that conform to the human body, thereby improving fit, comfort, and functionality in various fields such as fashion, clothing, and wearable technology. However, there are currently a limited number of approaches focused on this area. One example is Tactum [55], which utilizes on-skin gestures to design 3D models directly on the body. Another approach, ExoSkin [56], extends this by allowing for direct on-body fabrication using a custom handheld extruder. Both of these approaches are limited to creating passive geometries. In contrast, Zhu et al. [261] and Choi et al. [28] have presented devices for printing conductors directly on the skin, but they use a conventional CAD design-then-fabricate approach.

This thesis contributes to this line of work by presenting an interactive approach for direct on-body design and fabrication of epidermal electronics.

2.1.3 Concurrent and Bidirectional Fabrication

In particular, our approach is inspired by *concurrent* interactive fabrication, where design and fabrication are simultaneous and often controlled with a computer-assisted handheld tool. One notable example of this is the FreeD system [264], which utilizes a handheld milling tool to shape and carve 3D models with computer-assisted guidance. Other craft-inspired systems include Shaper Origin [179; 217], a tool for rectifying cuts on large-scale surfaces, allowing for user-directed precision cutting; and Enchanted scissors [254], which restrict the user to cutting only designated areas on a piece of paper. Additionally, Klamka et al. [105] developed a handheld dispenser tool for directly applying functional tapes on textiles to enhance fabrics with interactive functionalities. Recently, Tokac et al. [216] proposed a craft-inspired interactive fabrication system by integrating force feedback into a robotic arm for the clay carving process.

Furthermore, researchers have also explored the use of handheld tools for sketching and drawing. I/O Brush [185] is an augmented reality paintbrush that allows users to acquire textures, colors, and movements from the physical environment for immediate exploration and creation through drawing. Augmented Airbrush [198] guides users in the process of spraying a painting using a computer-controlled airbrush system, dePENd [253] provides support for sketching using pen and paper, and COMP*PASS [134] developed a compass-based digital drawing tool that integrates the advantages of digital control into manual sketching. While these systems provide handheld tools for interactive sketching, they are limited in terms of resolution and the creation of functional interfaces.

Bidirectional fabrication is another form of interactive fabrication that enables iterative manipulation of objects through digital and physical inputs [102; 235]. For instance, ReForm [234] presents a system that fabricates 3D objects based on on-the-fly modification of digital models and updates digital models after the physical deformation of objects. Similarly, this thesis presents bidirectional physical sketching tools that record manually sketched traces and print digitally modified designs.

This thesis contributes beyond prior work on interactive fabrication in three ways. First, it enables freehand on-body design and fabrication of epidermal interfaces. Chapter 3 presents the first computer-assisted handheld tool that augments freehand inking with digital support. Second, this thesis takes up the idea of interactively adjusting to be-fabricated designs in real-time and introduces the first high-resolution handheld printer for the in-place fabrication of electronic interfaces (Chapter 4). Third, this thesis combines the idea of real-time interactivity between humans and machines with the ability to print high-resolution marks and presents the first robotic printer that supports mixed-initiative sketching (Chapter 5).

2.2 Sketching Interfaces

Sketching, as a means of manifesting, sharing, and discussing ideas, has been widely acknowledged (e.g., [175]). However, manual skill is a necessary requirement for effective sketching. In order to make sketching more accessible, a plethora of sketching interfaces has been developed to enhance the fidelity of rough sketches. This section aims to provide an overview of prior research pertaining to both physical and digital sketching interfaces and drawing tools for 2D surfaces.

2.2.1 Physical Sketching Interfaces

In order to facilitate the creation of more accurate and detailed sketches, physical sketching practices can be augmented with visual and/or tactile guidance. For instance, the use of technologies such as Illumipaper [104] and Penlight [200] can be employed to highlight target areas, whereas systems such as Exoskin [56] provide visual guides for on-body fabrication. Additionally, several studies have investigated the use of screens to display additional information during the sketching process [49; 204; 205].

Unlike visual guides, tactile guides are only experienced while actively sketching. For example, the dePENd system [253] utilizes a ballpoint pen that is actuated by magnetic attraction to specific positions using a permanent magnet. Langerak et al. [111; 112] demonstrate the use of an electromagnet to create variable force and investigate algorithms to minimize tracing errors. Soheil et al. [100] employ friction-based haptic guides in Phasking Interfaces to explore how the sketching process can be shaped through shared control between the user and the system. We expand upon these ideas by proposing novel physical sketching tools for the creation of functional and high-resolution marks that can also roam autonomously.

2.2.2 Drawing Tools for 2D surfaces

Commonly used drawing tools include pen and paper. Previous work has explored various ways to make drawings at large scale and on arbitrary surfaces more autonomous and accessible. Examples of this include the use of XY pen plotters [14; 194; 219], hanging

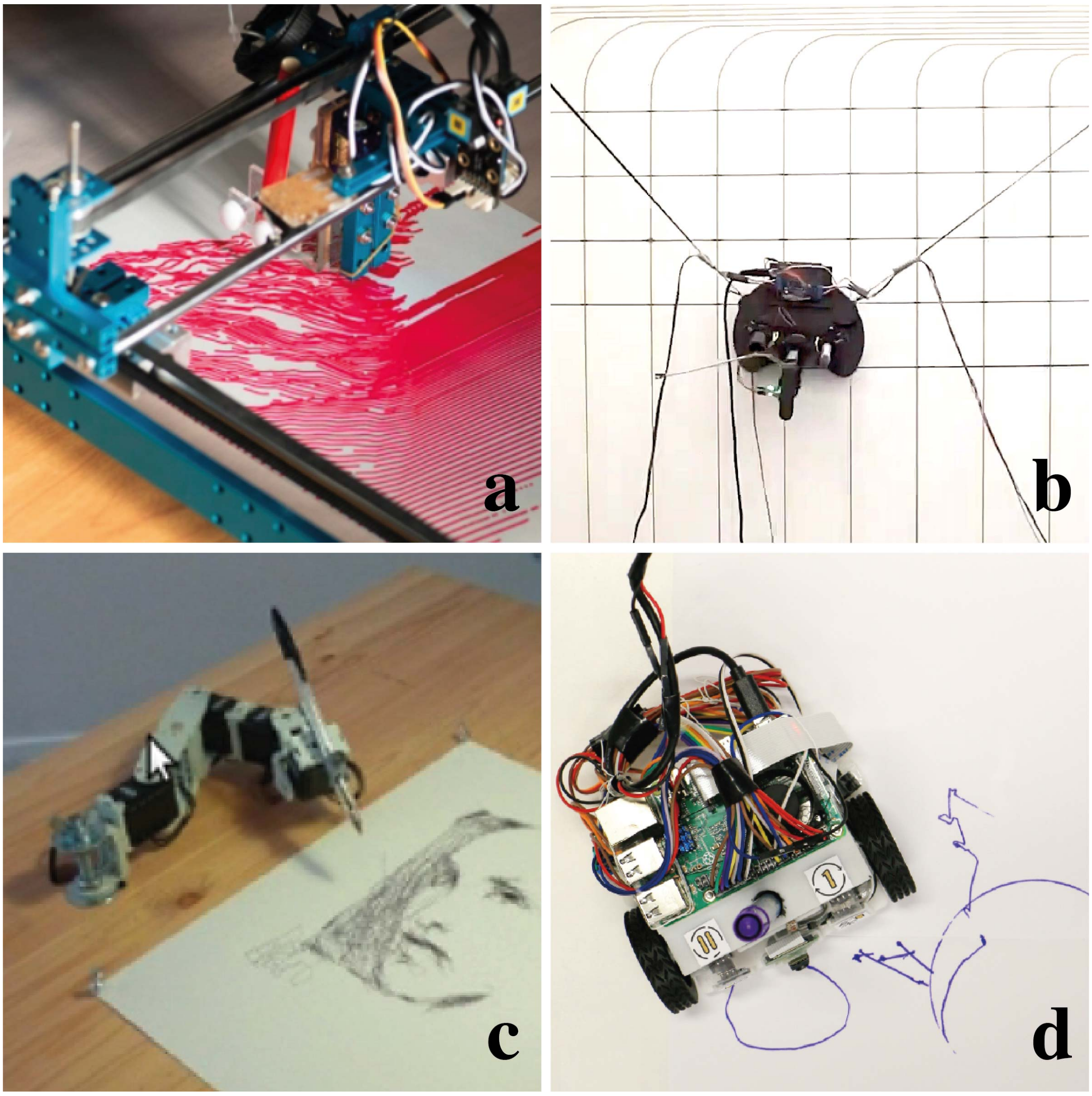


Figure 2.2 Examples of drawing tools for 2D surfaces include: a) #PlotterTwitter, an online community developing custom plotter for creating artwork [219], b) Duco, a hanging plotter for direct-circuit-writing on vertical everyday surfaces [25], c) Paul, an arm robot for drawing portrait [218], and d) Cobbie, a mobile robot that generates creative and diverse sketches [119].

V-plotters [25; 38; 141], or robotic arms [218; 249] to automate the drawing process on horizontal and vertical surfaces (Figure 2.2). However, these devices have some limitations, such as low printing speeds and the use of pens or markers, which restrict them to printing vector graphics. Inkjet heads have been proposed as an alternative to markers as they allow for printing high-resolution raster graphics at higher speeds, as demonstrated in

WallPen [228]. Nonetheless, these devices are still limited by the size of the device's drawing area, as they do not move freely.

To overcome this limitation, researchers have proposed using wheeled robots that can move freely and print on any size and shape of the surface. Lee et al. [114] introduced one of the earliest examples of sketching robots, and other examples include Cobbie [119] and DIY Omni Wheel Plotter [120]. Sustainabot [180] is a small robot printer that utilizes everyday materials to create shapes, and Kino [94] generates temporary patterns by etching fabrics. Additionally, there are several commercial sketching and printing robots available for education [86] and construction [181; 182]. In this thesis, building upon the existing research in the field, we propose an interactive fabrication technique that allows for on-the-fly modification of the design.

2.2.3 Digital Sketching Interfaces

Many interfaces for digital sketching have been developed, with a focus on either supporting users and enhancing their skills within digital environments or replicating the experience of sketching with pen and paper (see Figure 2.3). One early example of the former is SketchPad [212], which revolutionized traditional drawing by using a display and a light pen. Other notable examples include SILK [110], which adds interactive behavior to simple line-drawn interface elements, and PortraitSketch [251], which corrects digitally drawn lines to better match a predefined template. FlexStylus [50] is another example, which allows for adjusting parameters like stroke-width or color based on how a digital pen makes contact with a display. More recent developments include DesignScript [5], DressCode [89], and Dynamic Brushes [88], which aim to make drawing easier for users with a more intuitive interface. Additionally, other approaches have focused on providing guidance [115], tactile feedback [113], beautifying the strokes [82; 252], or enabling dynamic brushes and strokes [117; 231]. With recent advances in artificial intelligence, collaborative design with an AI agent has become possible, enabling iterative ideation [35; 143] and mixed-initiative content creation [39; 51]. Inspired by these works, we provide real-time assistance enabling a supportive and accessible environment for creative expression.

This thesis contributes to the field of physical sketching in three aspects. In Chapter 3, a sketch-based interface is presented to assist designers in drawing epidermal devices on-the-fly directly on the body. Chapter 4 introduces a physical sketching handheld printer that can simulate a multi-tip tool, allowing for the counteraction of user error and providing an alternative to tactile guides. Finally, in Chapter 5, the ideas presented in

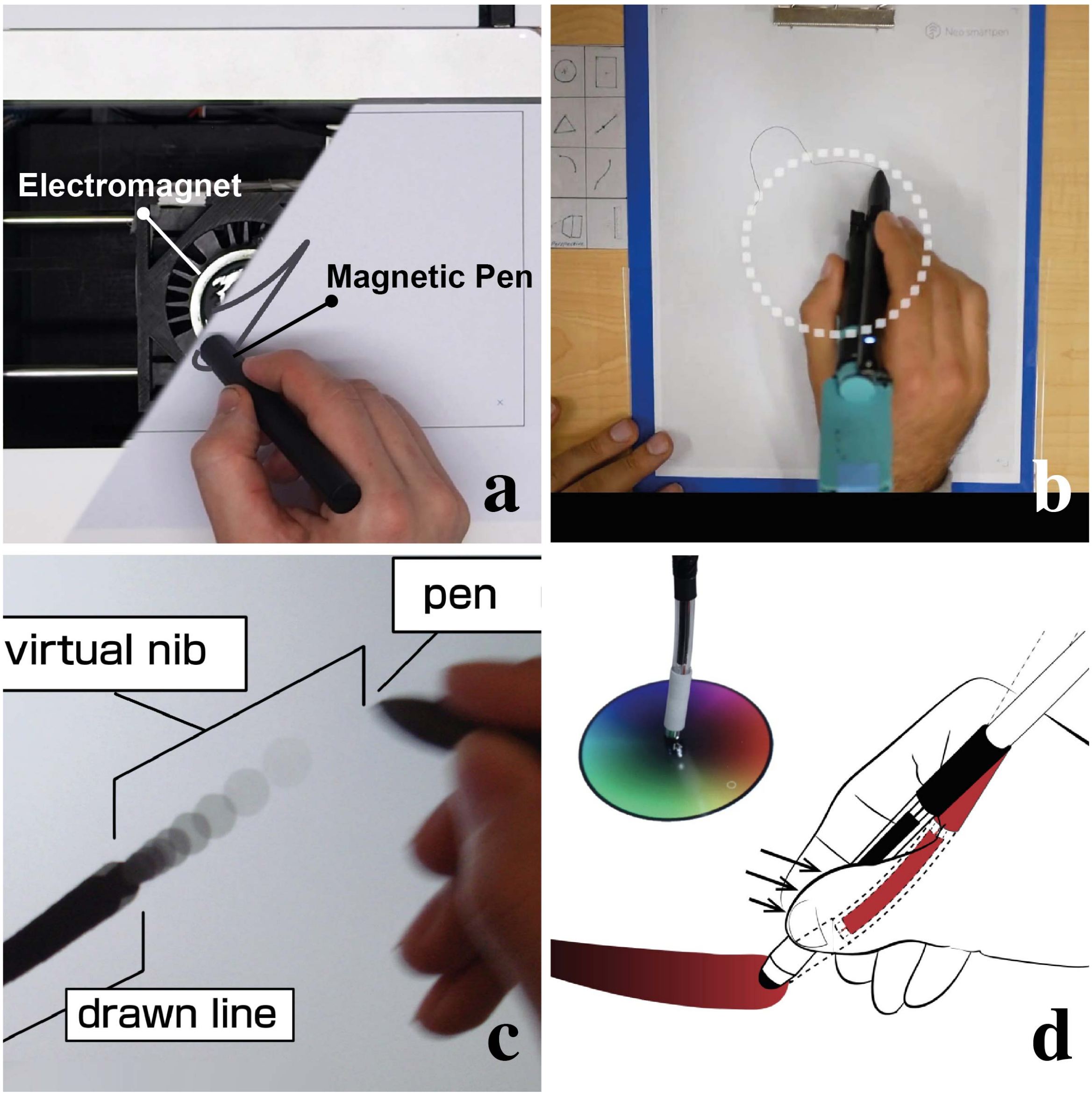


Figure 2.3 Examples of physical and digital sketching handheld tools include: a) A magnetic pen that provides dynamic guidance in sketching via a moving electromagnet [112], b) Phasking on Paper, a pen that creates force-feedback for physically assisted sketching [100], c) FlexibleBrush, a realistic brush stroke experience with a virtual nib [231], and d) FlexStylus, a flexible stylus that leverage bend input for improving the expressivity of digital art [50].

the previous chapters are expanded upon by linking sketches in the physical and virtual worlds through the use of a robotic printer on wheels.

2.3 Fabricating Electronic Circuits

Fabricating electronic circuits has traditionally been a complex and time-consuming process, requiring specialized equipment and expertise. However, recent advances in fabrication techniques have enabled the deployment of new interfaces in everyday environments with greater ease. This section will explore the various methods and techniques used in fabricating printed and flexible electronics, including inkjet printing, screen printing, and conductive stylus or paint application. Additionally, we will delve into the concept of sketching electronics and the use of epidermal electronic devices for interactive purposes.

2.3.1 Prototyping Printed and Flexible Electronics

Conventional approaches for fabricating customized interfaces with printed electronics include inkjet printing [20; 98; 99; 145], screen printing [148], plotting [25], and hydrography [60] (see Figure 2.4). Recent advances in fabrication techniques enable the deployment of new interfaces more easily in everyday environments. These methods and techniques involve applying sensors and actuators over existing elements and objects while preserving their original form-factor. For instance, they enable the deployment of interfaces on furniture, household items, and textiles, such as by attaching functional stickers [26] or ironing-on functional patches [208]. Furthermore, they bring interfaces to more flexible and sensitive surfaces, such as skin [28]. They can even enhance architectural structures by spraying functional graffiti [240].

On the other hand, to facilitate more rapid hands-on prototyping, researchers have proposed using a conductive stylus or conductive paint [17; 125; 136] or copper tape [189], which are directly applied on real-world objects. Prior research has highlighted the significance of aesthetics in circuit design [17; 169; 170]. However, digitally designed tools for flexible electronics [75; 144; 145; 173] have mainly focused on supporting novices with parametric electronic components to create electronic devices. In contrast to prior work, our approach draws inspiration from the direct hands-on fabrication of circuits and sensors using a handheld tool, and digitally supports users in creating visually aesthetic parametric components, which are directly designed on the object or body.

2.3.2 Sketching Electronics

Sketches are traditionally delivered physically on paper, however, nearly any medium is possible. Human-computer interaction (HCI) has a long tradition of exploring sketches

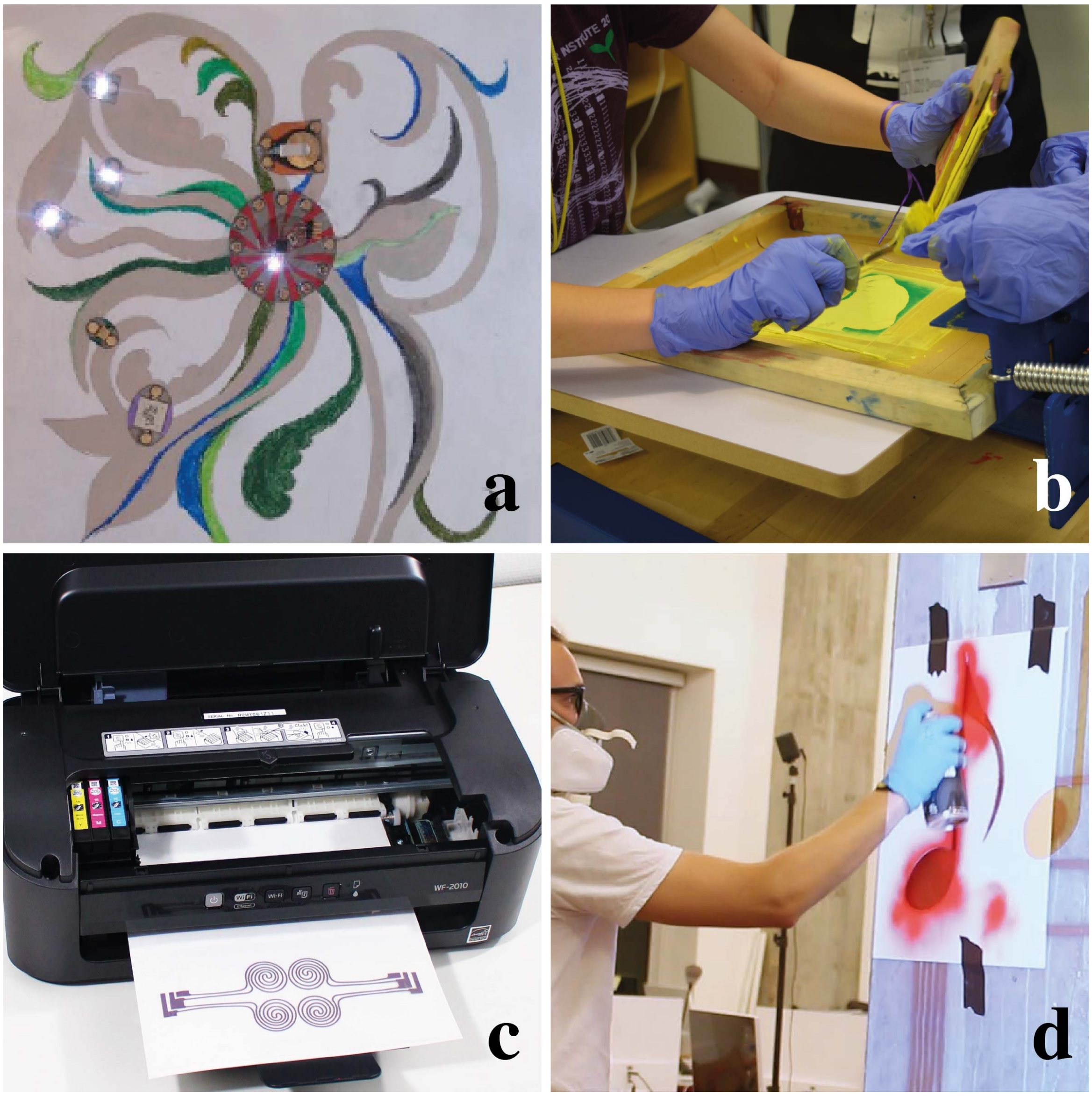


Figure 2.4 Examples of fabrication techniques for prototyping printed electronics include: a) conductive stylus or paint [17], b) screen printing [108], c) inkjet printing [99], and d) aerosol spraying [240].

that are spatially integrated within the real world [213]. While a product or graphical design can be sketched with pencil on paper, sketching an interactive product requires not only the implementation of its static appearance but also its dynamic behavior [77]. This means that electronic systems need to be sketched as functional devices, which is typically referred to as sketching in hardware. Mellis et al. described their approach to sketching in hardware as an Untoolkit [125]. Their strategy is to provide tools and techniques that allow existing components and materials to be used in new ways, rather than designing high-level components as kits such as the Phidgets system [83].

Follow-up work merged this approach with traditional craft. For example, Qi et al. extended this work by combining it with traditional papercraft [170], which led to the successful Chibitronics platform [27]. Buechley and Perner-Wilson [18] extended the idea of sketching in hardware to textiles. Together with Satomi, they presented a Kit of no parts, which shows how to build electrical prototypes from a diverse palette of craft materials [156]. To present the support of digital design in the creation of functional circuits, researchers introduced several design tools. For instance, Ellustrate [123] is a digital design tool that supports the functional and aesthetic design of electronic circuits using multiple materials and eases the task of practical circuit creation while guiding the users through the fabrication and debugging process, thereby enabling a new electronic design conversation. PaperPulse [173] is a design and fabrication approach that allows non-technical designers to create standalone interactive paper artifacts by augmenting them with electronics, using pre-designed visual elements and widgets to specify functional relationships between interactive components. PaperPulse generates layered electronic circuit designs and code that can be deployed on a microcontroller and provides assembly instructions for integrating the microcontroller and widgets in the final paper artifact.

While these approaches enable incremental and iterative sketching of interactive electronics in a broader range of settings than traditional methods, the fidelity of the resulting pieces is limited to what can be manually fabricated. In chapters 4 and 5, we present a method for combining the expressiveness of sketching with the precision and reproducibility of printing, using a hand-held printer.

2.3.3 Epidermal Electronic Devices

Building upon pioneering work in the field of epidermal electronics [103], the HCI community has proposed using on-skin devices for interactive purposes [237]. Various forms of input and output on the epidermis have been investigated, including multi-touch input [140], visual feedback [96; 122; 230; 239], haptic output [67; 69; 133; 246; 255], stiffness change [95], and customized physiological sensing [139]. Furthermore, fully wireless solutions for these devices have also been proposed [124].

The human body possesses a complex geometry with various recognizable landmarks [203]. These complex features of the body can be utilized to provide additional haptic guidance [239] and serve as mnemonic aids in interaction [12]. The significance of aesthetics in the social acceptability of body-worn interfaces has been acknowledged early on in research. Cosmetics and traditional body decoration have served as important sources of inspiration. For instance, Vega et al. [223] and Kao et al. [97] demonstrate how something as personal and customizable as makeup can be transformed into an interface. Other studies have presented approaches for the aesthetic design of on-skin electronics [122; 237] and functional tattoos [224]. In these studies, even though various levels of customization exist, the design process is separated from the body and occurs prior to fabrication.

Recent research has started to investigate the social acceptability of on-skin devices [256; 257], a theme closely linked to body art and tattooing [176]. Along these lines, input (or inking) on someone else's body [34; 207], as well as the impact of interactive body markings on the wearer's social image [106; 256; 257], are emerging areas of study. We do not explicitly aim to investigate social issues surrounding interactive body markings, however, the created artifact in Chapter 3 could serve as a tool to further investigate on-body co-creation and facilitate the design of socially acceptable on-body interfaces, by making the chosen body location (c.f., Harrison et al. [71]) more directly apparent during the creation process.

2.4 Touch Sensing

Touch is a widely used input technology, with a long history dating back to the popularization of screen-based devices [10]. Since then, it has evolved and been applied in a diverse range of areas and scales, such as interactive spaces [42; 260], objects [147; 188], and on-body interfaces [70; 73; 140; 237; 239]. The field of custom touch sensor fabrication has also seen a wide range of methods developed, including crafting with conductive copper and gold leafs [96], silicone casting [237], inkjet printing [98], screen printing [140; 239], and 3D printing [190].

A variety of technologies can be used to sense touch, including optical methods such as frustrated total internal reflection (FTIR) [68] and depth cameras [70; 244], commonly used for large-scale touch screens. Acoustic methods have also been demonstrated in touch interactive surfaces [126] and on the body [72; 73]. Other technologies include resistive methods [76; 237], electric field sensing [262], impedance profiling [188], time-domain reflectometry [245], and electric field tomography [259].

Projected capacitive sensing is one of the most widely accepted and frequently used methods for sensing touch [59; 61; 63]. In particular, mutual-capacitive sensing has gained popularity due to its ability to enable high-resolution sensing of multiple simultaneous touch contacts, its ability to be embedded in small form-factor devices [63; 178; 250], and its low-latency [116]. Despite its widespread use, prototyping multi-touch applications requires advanced knowledge of the underlying technology and specialized hardware. In this section, we first provide an overview of the research on prototyping touch-based interfaces. We then discuss related work on different touch-sensing modes and available commercial touch controllers.

2.4.1 Prototyping Tools for Touch-Sensing

With the increasing popularity of camera-based methods for multi-touch sensing, various software frameworks have been introduced to simplify the process of prototyping and implementing touch-sensing applications. For example, camera-based systems such as ReacTIVision [93] have stimulated extensive research in the field of multi-touch sensor surfaces. More recent work, such as the depth-camera-based RoomAlive, incorporates modern sensing methods into accessible toolkits [90].

Similarly, capacitive sensing toolkits have had a similar impact on capacitive touch sensing. The *CapSense library*[1] is an Arduino library for loading mode capacitive touch sensing and does not require specialized hardware or advanced knowledge. However, it is limited by the inherent drawbacks of loading mode sensing and cannot deliver high-resolution or multi-touch sensing without complex instrumentation. The *OpenCapSense* toolkit [61] is more powerful and supports several forms of capacitive sensing, including mutual capacitance sensing. However, it is primarily designed for hover and gesture recognition, lacks the capabilities to support high-resolution touch sensor surfaces, and requires a proprietary hardware controller board. In contrast, our proposed approach (see Chapter 6) utilizes a commodity microcontroller.

In addition to hardware-related approaches, software frameworks are available for processing and classifying multi-touch input for interaction. An overview of these frameworks can be found in [41]. Our proposed firmware and software libraries in Chapter 6 draw inspiration from this prior work and propose a novel solution for rapid prototyping of capacitance-based sensing for high-resolution multi-touch input.

2.4.2 Sensing Modes and Commercial Touch Controllers

In the field of capacitive sensing, there are several different sensing modes that have been developed and used in commercial touch controllers (see Figure 2.5). The simplest mode is the loading mode, which can be easily realized by electronics novices using the Arduino CapSense library and adapted to custom designs. Several commercial controllers are available for loading mode sensing, such as the MPR121, Microchip MTCH6102, and

[1] https://playground.arduino.cc/Main/CapacitiveSensor

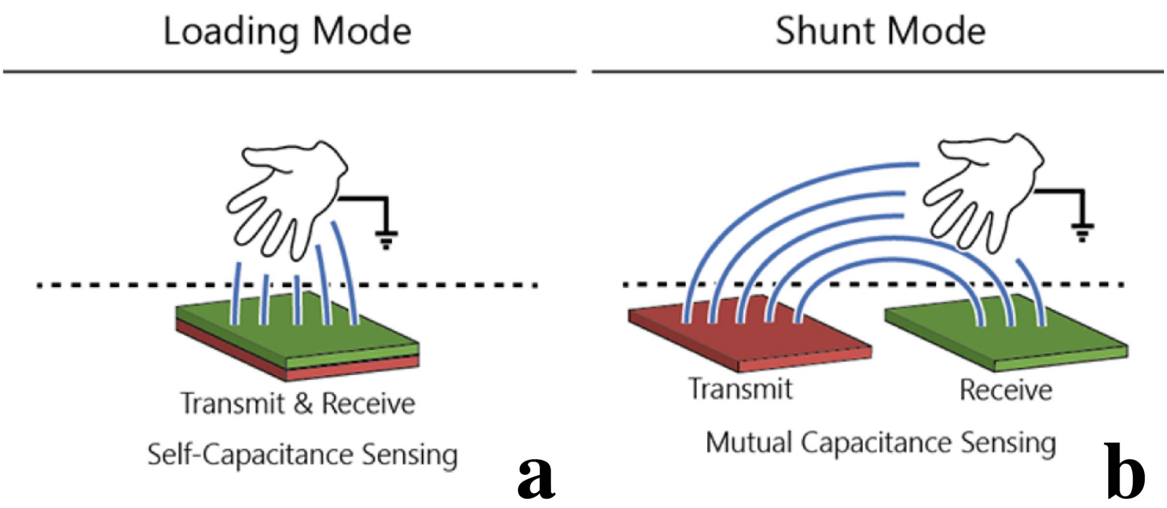


Figure 2.5 Capacitive sensing techniques: a) loading mode [63] and b) shunt mode [63].

Analog Devices AD7142 [63]. However, this technique is low resolution and limited to single-touch detection.

In contrast, most multi-touch sensing approaches require complex hardware. For instance, shunt mode (also known as mutual-capacitance (mCap)) sensors measure the change in capacitance between two intersecting conductors caused by the proximity of an external conductive element, such as human touch. To measure this change in capacitance, mCap controllers transmit an AC signal through one electrode (TX electrode) and observe the received AC signal at the other electrode (RX electrode). Touch-sensing surfaces are created by organizing these TX and RX electrodes into a row-column matrix. However, in these sensors, the change in mutual capacitance between touched and not-touched states is typically much smaller than the stray capacitance [36]. Therefore, sophisticated measuring methods, such as Capacitance-to-Digital Conversion (CDC), Sigma-Delta Modulation, and Successive Approximation with Single-Slope ADC [197], are required to identify touch input with a sufficient signal-to-noise ratio. These methods require complex analog circuitry and cannot be implemented on commodity microcontrollers, such as an Arduino, without specialized hardware.

Dedicated controller chips that implement sophisticated measuring methods for capacitive sensing, such as Microchip MTCH6301 and Texas Instruments MSP430FR2xx family, are commercially available. While these chips may be suitable for electronics experts, they may present significant challenges for novices, interaction designers, and makers in terms of programming, adaptation, and interfacing.

In conclusion, while commercial touch controllers can serve as viable options for professionals in the field of electronics, they may pose significant challenges for individuals with limited experience in electronics and programming. This can be attributed to the fact that such controllers come with specific instructions for sensor designs, such as the precise dimensions of electrodes, which can constrain the flexibility of customized sensor designs. Furthermore, designing custom sensors may impact the underlying signal models and necessitate novel circuit designs for accurate touch sensing. To effectively overcome these challenges and facilitate sensor read-out on customized sensate surfaces, realized with sketching tools and techniques presented in Chapters 3-5, a do-it-yourself multi-touch scheme is proposed in Chapter 6.

3 BodyStylus: Freehand On-body Fabrication

As highlighted in the introduction, creating functional interfaces on complex geometries such as skin presents significant challenges. The growing interest in skin interfaces is due to the appealing properties of skin as an input surface. Skin is readily available, large, soft, and deformable, enabling expressive interaction beyond touch [238], and its tactile perception facilitates eyes-free interaction [64]. Prior research in and beyond HCI has explored a multiplicity of materials, fabrication techniques, and design approaches for exploring these properties in epidermal interfaces. They typically follow a three-step workflow: (1) designing the interface on a computer, e.g., using graphic design software [96; 122] or custom software tools [124; 140]; (2) fabricating the design, e.g., using screen printing [122], stencil or laser cutters [96; 124; 237], inkjet printers [99] or lamination techniques [124]; and eventually (3) applying the epidermal interface to the body, e.g., through water-transfer [122]. Recent work by Choi et al. [28] succeeds in merging steps 2 and 3 into one unified on-body fabrication step using an attachable printing device and conductive ink. Nevertheless, all of these approaches require an initial digital design step, separated from the body. In contrast, we wish to support the hands-on adaption of designs to the human body as is common in established body-based practices including tattooing, make-up, and henna.

Such direct on-body design enables fitting epidermal devices to the human bodyscape, e.g., making fine adjustments according to detailed shape and movement of the body, and supports in-situ creative exploration and expression. Pioneering work on computational on-body design and fabrication has so far been restricted to the fabrication of passive designs without interactive functionality [55; 56]. The addition of functional elements, however, is particularly challenging due to the complex interplay between circuitry,

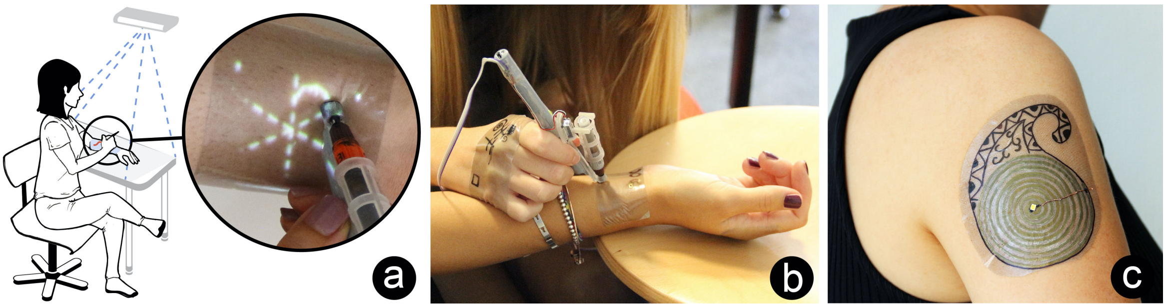


Figure 3.1 *BodyStylus* combines the design and fabrication of functional on-body interfaces into one integrated activity performed directly on the body. A handheld device combines the ease and directness of free-form drawing with digital assistance; visual cues provide guidance while constraints dynamically restrict inking to prevent errors. *BodyStylus* enables drawing on one's own body (a), as well as collaborative design (b), to realize functional epidermal interfaces (c).

bodyscape, and aesthetics. Techniques from traditional on-body arts and crafts underpin the importance of a tight link between the design and the individual body size and shape. With this in mind, we aim to explore the combined design and fabrication of *functional* epidermal devices directly on the body.

In this chapter, we address these challenges by presenting *BodyStylus*[1], the first computer-assisted approach for freehand on-body design and fabrication of epidermal devices. *BodyStylus* consists of a general concept, inspired by traditional body-art, a system including a handheld marker (Figure 5.1a), and a set of design techniques for creating epidermal interfaces. The handheld marker enables on-the-fly design, customization, and fabrication of epidermal interfaces directly on the user's body. The system supports the user in implementing functional devices respecting electronic constraints, aesthetics and body features: projected in-situ guidance facilitates functional and aesthetic results, while automatic switching between inking and non-inking modes prevents errors in circuit logic. We demonstrate the technical feasibility of *BodyStylus* with a proof-of-concept implementation using a custom-designed dispensing pen and self-sintering conductive ink to instantly create functional traces on epidermal devices.

The design techniques offer the user a rapid, direct, and hands-on way of creating visually aesthetic epidermal devices on the body. Based on varying levels of guidance and constraints, we offer techniques for inking functional conductive traces, creating aesthetic shapes and patterns, adding electronic components, and creating custom free-form components.

We show how these techniques can be used in concert to design and fabricate functional epidermal devices, alone or collaboratively, for various body locations. We demon-

[1]This chapter is based on a publication at CHI'21 [162]. As the first author, I led the development of the fabrication technique, created the demonstrators, and conducted the evaluations.

strate that *BodyStylus* reflects aesthetic and artisanal paradigms of traditional on-body arts and crafts, while also offering a means of facilitation and guidance orthogonal to traditional crafts.

In two focus groups, *BodyStylus* was practically explored with engineers and make-up artists. The findings uncover commonalities and differing perspectives as well as suggestions on how the practitioners would wish to incorporate it into their own workflow. The aspiring make-up artists were able to implement epidermal controls for special-effect LEDs within less than two hours after having learned about the concept of epidermal devices. Our observations further show that working on the body inspired critical reflection on the relationship between bodyscape, interaction, and design.

3.1 Design Context

In this section, we take a close look at off-body and on-body work practices of established arts and crafts, and describe the design opportunities for *BodyStylus* that result from the unique combination of aesthetics, human bodyscape, and circuit logic.

3.1.1 Draw on Aesthetically Rich Traditions

Body markings are a time-honored and recurrent motif in the history of civilization: ocre body-paint on ritual performers, which might be considered one of the first expressions of human art, was used as early as 147,000 years ago to augment faces and bodies [232]. Permanent tattooing [54] and henna markings [149] have a long and complex history. Contemporary body markings include permanent tattoos of various styles [167], various forms of the non-permanent face and body paint, e.g., cosmetics, stage or carnival make-up, as well as semi-permanent drawn-on skin embellishments [84] from plant-based dyes, e.g., henna or jagua.

What many of these traditions share is that they work with hand-held tools directly on the body and that the tools they use shape the aesthetics of the body marking and vice versa. For instance, depending on the desired aesthetic, a tattoo might be created using a tattoo machine vs. using so-called hand poking. In henna art, aesthetics evolved around what the available tools and materials afforded. Varying pressure applied to the henna applicator affects the geometry of the resulting shape, e.g., by creating a droplet. This tight coupling of the hands-on tool, ink, and skin contrasts with existing practices around on-skin devices, such as electronic tattoos, which are mostly designed and fabricated off-skin and only then applied.

BodyStylus builds upon these rich traditions of elaborating a design on the body by enabling the user to *generate circuits using a pen on skin.* Thereby, the process of fabricating on-body interfaces becomes more immediate and less detached from the body than CAD-based approaches. In its aesthetic, *BodyStylus* strongly draws on the intricate line art of henna, which, with its crisp lines and entwined ornaments is well suited for implementing electrical circuits.

3.1.2 Relate Bodyscape, Ornament and Function

Many traditional on-body arts and crafts incorporate the individuality of the human body into their practices: designs are adjusted to the unique size and shape of the body – often as they are applied. For instance, tattoo artists would transfer a tattoo stencil onto the desired body location, and then iteratively tune body location, size and orientation in communication with the client. If corrections are needed, they would remove and re-apply the stencil, or sketch adjustments using sterile skin scribe markers [225]. Characteristically, templates used in maternal henna are often round and symmetrical, but the mother's belly is usually not. During application, the henna artist would then adjust the design on-the-fly by correcting for the belly's size, e.g., by adjusting the ornament's symmetry to fit the belly button's position [168]. In contrast, current design and fabrication techniques for epidermal interfaces do not afford such on-body adjustments to the individual bodyscape.

In addition, a marking's function can both follow and dictate its body location. Many types of body markings serve functions reaching beyond purely aesthetic or ornamental purposes. Traditional henna markings act as luck or fertility charms and are used in folk medicine [149]. In these applications, the marking's location is, for instance, determined by its healing purpose: archaeological evidence hints that ancient healing tattoos (e.g., found on 'Ötzi') overlap with acupuncture points found in Chinese medicine [43; 107]. More recently, permanent 'Medical Alert Tattoos' have gained momentum, e.g., to alert first responders to chronic conditions such as diabetes [24; 109]. Contemporary henna markings may also serve to indicate marital status [187] or as a sign of having participated in elections [92], and even for advertisement [174]. In these cases, markings are placed where good visibility and immediate discoverability can be achieved.

Similarly, the on-skin placement of modern sensing technology follows body landmarks [239]: sensors might be placed where they achieve accurate measurements and avoid stigmata, haptic actuators where tactile acuity is largest, and display elements where they are highly visible. As a result, bodyscape and circuit function form a two-way relationship, where characteristics of the body motivate choice and placement of circuit elements, and vice versa. In consequence, unlike e.g., paper circuitry design, on-body design needs not only to follow "electrical, material, and visual design principles" [123], but also adhere to design principles imposed by the user's bodyscape.

BodyStylus supports the designer in on-the-fly customization of designs, directly on the body, to fit the individual bodyscape they want to apply it to. This way, *BodyStylus* achieves compliance with the inherent diversity of human bodyscapes and allows matching body location to interface function.

3.1.3 Mind Constraints from Geometry and Circuit Logic

Electrical circuits contain a layer of invisible information, i.e., their inherent logic and physical rules, for instance, polarity and continuity. In consequence, circuit construction is challenging to novices, designers, and even hobbyist makers, as it involves the risk of violating electronic design rules (e.g., miswiring, short circuits) [13; 123], erroneous component selection [13], and creating functional errors [123]. When drawing circuits on skin, these issues intensify: all traces might look the same, but their functions are not. Some are essential to keep the circuit intact, some have to be isolated from each other to avoid short circuits, while others might be purely ornamental and open to artistic freedom. Combining circuit logic and artistic design increases the complexity of the circuit, making it harder to understand and debug [137]. In addition, the resistance of the circuitry can depend on the thickness of ornamental traces and on the type of conductive ink used - a characteristic which designers might find challenging [2].

BodyStylus implements free-form drawing and allows for artistic expression, but also ensures circuit logic and aesthetic qualities, such as symmetry or harmonic repetition. To achieve this, *BodyStylus* continuously adapts to the user's input by physically preventing mistakes that would compromise circuit logic (e.g., avoiding miswiring), or by dynamically responding to adaptations introduced by the user (e.g., change of scale).

3.1.4 Use Guides for Planning and Facilitation

There are aesthetic elements that are difficult to "get right" in free-form drawing, e.g., circular, repetitive, and evenly spaced shapes or straight lines. Many traditional on-body arts and crafts apply guides to facilitate complex shapes while finalizing and perfecting designs on the body. In tattoo art, the creation of stencils is considered an art on its own [225; 241]. While some freestyle tattoo artists despise the use of stencils as amateurish [47; 196; 225], thermal transfer stencils are today widely used to provide an outline for the tattoo artists to fill in. Similarly, off-the-shelf templates are rarely used by experienced henna artists, who instead rely on skill and muscle memory. Nevertheless, guiding tools and techniques also exist in henna art: for large henna ornaments, where spatial division is crucial, motifs are typically planned by "dotting" or outlining large elements first, and filling in details later. Guidance grids can ensure even spacing [33], and imprints from cookie cutters facilitate the free-hand application of round or symmetric shapes [186].

Stencils as guidance tools have limitations. They negatively impact the crispness of lines in make-up art and can be distorted when applied to cylindrical or convex body shapes. In addition, some guidance techniques are not directly applicable to skin. For instance, henna artists would typically practice shapes and patterning techniques on an acrylic sheet with a template underneath. When later creating henna art on skin, the artist can recall the practiced shapes or techniques from muscle memory without requiring additional guidance. This highlights how guides are useful in the learning process, specifically in skill-building activities, which has also been shown by prior work in HCI [191]. In addition, some types of guides (e.g., plastic henna stencils) confine the skin patch where ink or paste can be applied to. For instance, by creating a physical barrier. Yet, the use of more restricting or proactive guides, while known in HCI (e.g., for error prevention as in [266]), is rare in traditional crafts and underexplored in the area of on-body art.

BodyStylus enables temporary sketching of circuits, patterns or shapes, and allows using guidance marks. The users can iterate upon these virtual sketches. Once they are satisfied with their sketch, they can switch to inking mode and elaborate on the actual circuit re-using the sketch as guidance and applying constraints where needed.

3.2 BodyStylus

BodyStylus supports designing and implementing epidermal interfaces directly on the body. *BodyStylus* consists of a general body-art-inspired concept, a system including a handheld marker (Figure 5.1a), which enables on-the-fly design, and a set of design techniques. We now present the implementation of the *BodyStylus* system and introduce projected guidance and dynamic constraints as two core features to assist the user in creating epidermal interfaces that are functional, aesthetic, and customized for an individual's body.

3.2.1 Implementation

BodyStylus. shown in Figure 3.2a and 3.2b, uses a position-aware handheld pen that can proactively switch between different modes – inking and non-inking – and that supports

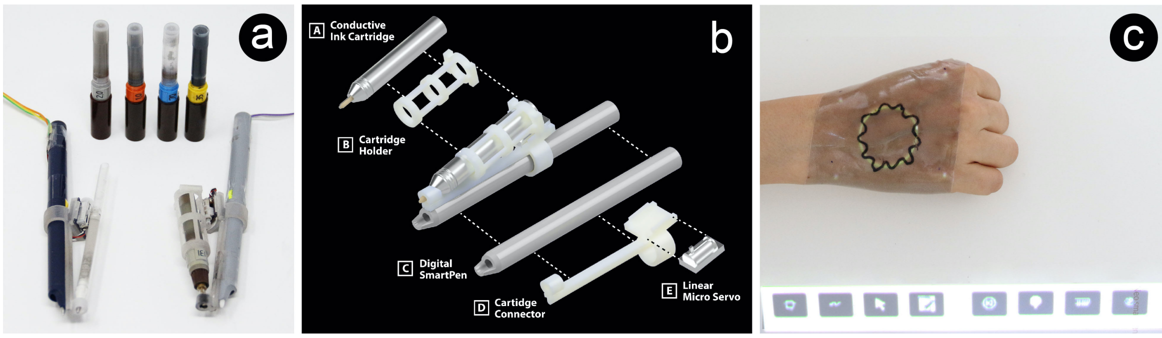


Figure 3.2 The handheld device includes a smart pen, multiple pen tip sizes, a refillable cartridge with conductive or non-conductive ink or conductive ballpoint pen, and a servo motor. (a-b) The motor can retract the ink pen to prevent inking. Extended functionalities are accessible on the user interface menu (c).

the fabrication of both conductive and non-conductive traces. The choice of a handheld inking device builds upon tools used in traditional body art. It allows for rich practices of designing and fabricating on and with the body, individually or collaboratively as a social activity.

Position Tracking. As a rapid means of prototyping high-resolution position tracking, we opted for using a smart pen relying on printed patterns [152]. For this purpose, the current prototype is based on a Neo SmartPen M1 [199].

Tattoo Substrate. A temporary tattoo patch insulates the electrically functional parts of the body and provides the smart pen with a trackable pattern. The pattern is laser printed onto a patch of temporary tattoo paper [150]. The tattoo paper is covered with a layer of impasto gel [57] (~0.15 mm). The gel prevents excessive distortion of the trackable pattern by slightly evening out the skin surface while preserving the ability to conform to the shape of the body. The finished substrate is then reversed and adheres to the body using tattoo adhesive. The gel also has the secondary benefit of providing structural strength to the epidermal device, allowing it to be removed and re-applied multiple times. This substrate is a pragmatic choice to support the current hardware setup. In the future, systems building upon *BodyStylus* might use different substrates or operate directly on the skin, e.g., using a different position tracking technology and an insulating spray primer.

Modular Ink Cartridges. A 3D-printed housing was attached to the Neo SmartPen (Figure 3.2b). The housing fit the Rotring isograph pen tip and ink-reservoir [87] to enable swappable cartridges. Cartridges with diverse non-conductive inks are available off-the-shelf, while empty cartridges can be filled with conductive ink. We use sintering-free nanoparticle gold ink [177] (~352 Ω/cm for 2 mm wide trace on our substrate). Pen tips of different sizes allow for controlling the thickness of ornamental and functional traces

(Figure 3.2a). We created a customized version of the housing to hold a commercially available conductive silver ink pen [195]. Due to its aesthetic appearance, we used gold ink as our default, however, for application scenarios requiring highly conductive traces, we used silver ink (~13 Ω/cm for 1 mm wide trace on our substrate) or a combination of gold and silver inks. To increase the conductivity of functional traces we applied a primer spray that has proven successful to improve ink adhesion [166] on the painted traces.

Constraints Through Retractable Inking Tip. To prevent invalid electrical connections, the pen features a computer-controlled retraction system. The modular cartridge is connected to the pen via a linear servo motor [4], controlled by an Arduino Uno microcontroller. This allows for computer-operated lowering or retraction of the inking tip (Figure 3.2a): touching on the body, the pen delivers ink, whereas when retracted, it enters non-inking mode.

Projected In-situ Guidance. *BodyStylus* assists the user in fabricating epidermal devices by displaying visual cues directly on the body. Our prototype uses a ceiling-mounted projector (Sony VPL-HW50ES) for a stationary setup. A portable projector (DELL M115HD) is used for displaying guides on body locations that cannot receive projections well from the ceiling, such as the upper arm or shoulder. A simple calibration routine is implemented by marking four separate points on the substrate corners and then using projection mapping on these points. This helps the user to keep the body in the correct position during the design and fabrication process. Alternative options for future implementations include visual augmented reality, e.g., realized using a head-mounted display, or a pico projector built into the pen [200].

***BodyStylus* User Interface.** In addition, *BodyStylus* includes a user interface that establishes a logic link between all of the above components. It comprises all system logic (e.g., switching between inking and non-inking mode), controls the projected visual guidance, and displays additional visual feedback and controls to the user. While most interactions are performed directly on the body, some extended functionalities (e.g., selecting electronic components and advanced visual styles) are accessible using projected buttons (Figure 3.2c). The Neo SmartPen sends position updates to a C# host application using Bluetooth. The C# application uses OSC to pass coordinates to the *BodyStylus* UI, implemented with JavaFX. The UI communicates with an Arduino microcontroller using the Arduino-serial-connector library to control the servo-motor which controls the inking tip.

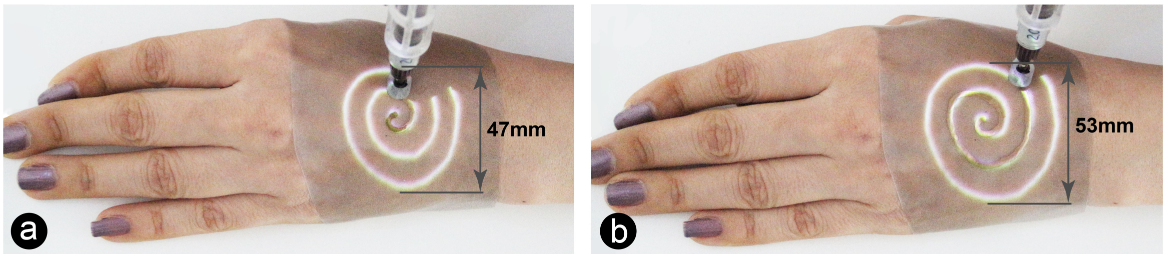


Figure 3.3 Visual Guidance. (a) In this example, visual cues assist the user in creating a spiral motif. (b) Guides adapt in real-time to user input. As the user deviates from the proposed trajectory, the design is continuously updated.

3.2.2 Assistance Through Guidance and Constraints

BodyStylus considers three high-level design parameters: the aesthetics of visual patterns, the bodyscape of the user, and circuit logic. This creates a demanding design space. Users who are experts in all three areas may create a functional and aesthetic on-body circuit by free, unassisted drawing with a conductive pen. However, most users will need assistance in at least one, or multiple areas. *BodyStylus* combines the ease, directness, flexibility, and creative power of freely drawing on the body with various levels of assistance through two key concepts that underlie our design: *guidance* and *constraints*.

Guidance. *BodyStylus* guides the user with visual cues projected on the body. Guides assist the user in getting the logic of the circuit right. E.g., while the user is drawing a conductive trace, guides may suggest a routing or may highlight terminals to connect to. Similarly, aesthetic elements might be difficult to get right in the free-form drawing, e.g., repetitive or evenly spaced shapes, as well as detailed ornamental features might be difficult to draw consistently. Here, visual cues are used to guide the user in optimizing their free-hand work. Furthermore, guides help the user in choosing valid body locations for components that depend on a specific positioning on the body.

To support improvisation during the inking process, guides can be set to adapt continuously while the user is sketching (see Figure 3.3). Hence the user can decide at any point to follow a design suggestion or not. In the latter case, the visual guide is updated to be in line with the modified design.

The level of detail of guides can be adjusted to adapt to the user's expertise and the desired degree of creative design freedom. For instance, visual guides can simply project outlines or directional indicators but if desired can include full details of a design, e.g., the detailed routing of a trace or the full visual pattern of an ornament.

Constraints. To prevent the user from inadvertently violating constraints relating to circuit logic, aesthetics, or body location, the pen can automatically switch between an inking

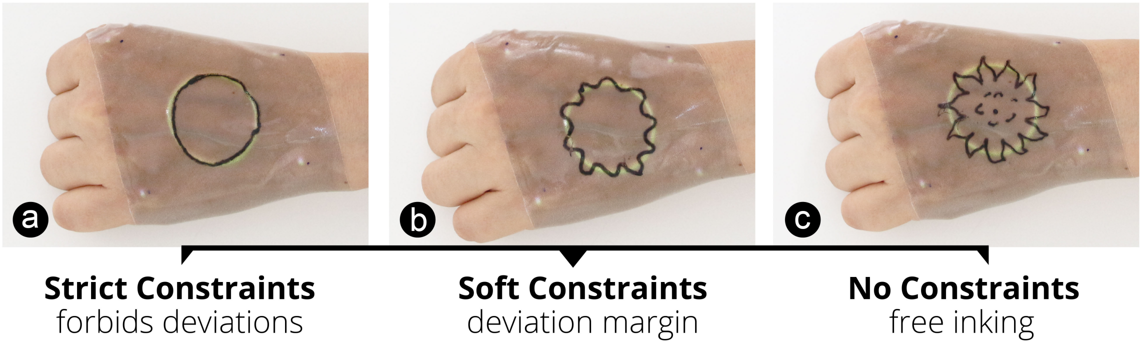


Figure 3.4 Constraints can be imposed at different levels. In this example, a strict constraint forces the user to closely follow the projected shape primitive (yellow circle, a). A soft constraint allows the user to deviate within a certain margin, for creative adjustments (b). Without constraints imposed, the user can draw freely, while still benefiting from high-level visual guidance (c).

mode and a non-inking mode. This enables the system to physically restrict the user's inking actions. For instance, when the user is about to create a short circuit, the pen can automatically stop inking by moving the tip of the cartridge up.

To free up creativity but provide safe boundaries for flexible exploration, the system allows the user to suspend constraints with a "manual override" (c.f., Zoran et al. [266]). In this case, the user repeats the desired action a second time to override the constraint.

Despite the pen only offering two distinct interactive modes (inking and non-inking), we define the constraint space as a continuum that allows more or less tolerance. At the strictest, it prevents any inking that was not planned out by the initial design (e.g., deviating from an original aesthetic). At the lowest level, it does not restrict the user's actions at all. Between the two extremes, it allows deviations within a certain margin, as defined by the user (see Figure 3.4).

Guidance and constraints can be flexibly combined throughout the design and fabrication process. This creates a two-dimensional space for interacting with *BodyStylus*, visualized in Figure 3.5. Depending on the current design task and her expertise, the user may benefit from more or less guidance and more or less constraints, for working on circuit logic, aesthetics, and bodyscape.

3.3 Design and Fabrication Techniques

We now present a set of interaction techniques for on-body design and fabrication with *BodyStylus* and describe how these are shaped by constraints and guidance (see also Figure 3.5).

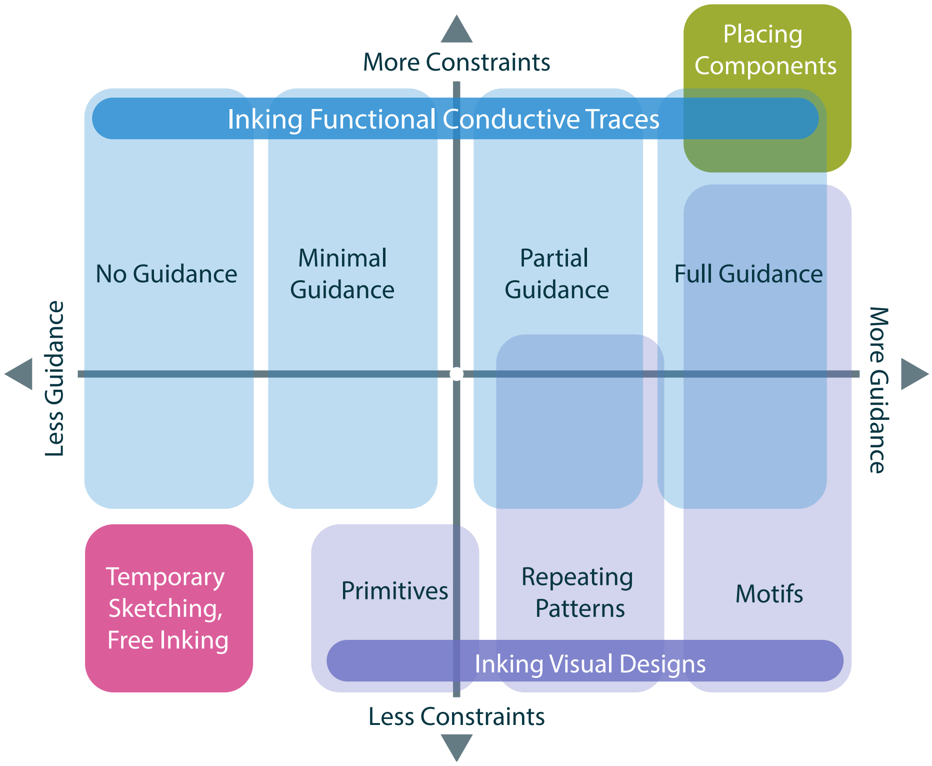


Figure 3.5 Guidance (x-axis) and constraints (y-axis) are orthogonal and complement each other during the fabrication process. While crafting a complete epidermal device, a user might flexibly move within this two-dimensional space depending on the current task. Free-form components combine design methods from inking conductive traces (blue), placing components (green), and inking visual designs (purple).

3.3.1 Temporary Sketching and Free, Permanent Inking

Before any part of the design becomes permanent, one can use the pen to draw virtual lines in the *non-inking mode*. These might be used to sketch an outline of the design or to experiment with how lines might complement the shapes of the body. As these virtual lines are not physically inked but merely indicated using visual feedback, users can move, scale, delete, and redraw as they wish. If one is certain of the pattern one wishes to place on the body, *BodyStylus* supports drawing of conductive and non-conductive traces by switching to the *inking mode*.

3.3.2 Inking Visual Designs

In inking mode, *BodyStylus* offers *primitives* as visual guides for artists to develop their designs, *patterning tools* to support artists in creating repeating patterns, and *motifs* for adding specific predefined elements to the design.

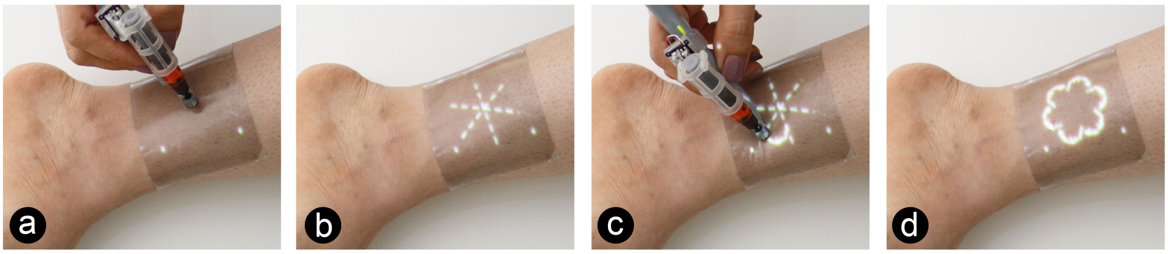


Figure 3.6 Radial symmetry tool enabling rapid designs of repeating patterns, building on practices from henna art. The user positions the center of the symmetry axes (a), and these axes appear as guides (b), the user draws the geometrical pattern in one sector (c), and the pattern is replicated in the other sectors (d).

Shape *primitives* such as line, circle, and spiral provide guidance to orient or outline the design. For instance, the user can select the circle tool, then specify its center and radius with a stroke to use it as a guide to drawing a flower.

Repeating patterns are an important characteristic of henna art, which we draw on for aesthetic inspiration. We provide visual guidance for creating multi-axial symmetries, repetitions of patterns, and fractal patterns. As an example, to draw a complex pattern with radial symmetry, the user selects the *radial symmetry* tool and locates the center point on the canvas (Figure 3.6a). The user selects the number of radial axes and (Figure 3.6b) sketches the desired design in one of the sectors (Figure 3.6c). The pattern is then projected in all other sectors for the user to trace (Figure 3.6d).

Finally, the user can select to transfer *motifs* to the body. The user has the option to select one of the available patterns (e.g., horseshoe or curlicue) or upload an SVG line graphic. This shape is then projected on the body at a location specified by the user. Here the system expects the user to trace the projected lines. If the user deviates from the projected line, the system scales and rotates the shape to allow the user to dynamically adjust the shape through drawing (Figure 3.3). Constraints can be used to limit the amount the user can deviate from the predefined shape.

Here, guidance is consecutively increased. *Primitives* are merely used as helper lines and remain unaffected by the user once placed. Tools for *repeating patterns* allow the user to create detailed templates – the system assumes that the user will trace this, but it is not enforced. Finally, *motifs* assume the user will always trace them, even dynamically updating to adjust to the user. In the dimension-space on Figure 3.5, we move from left *primitives* gradually further right *repeating patterns* and further up for *motifs*.

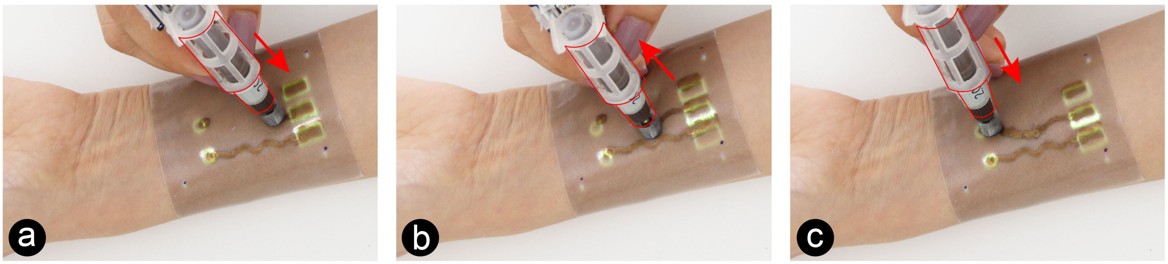


Figure 3.7 Constraints prevent invalid connections of conductive traces. The user draws a conductive trace to connect two terminals (a). As she is about to connect to another conductive trace and create a short circuit, the pen retracts to non-inking mode (b). Moving the pen away from the original trace switches the pen back to inking mode (c).

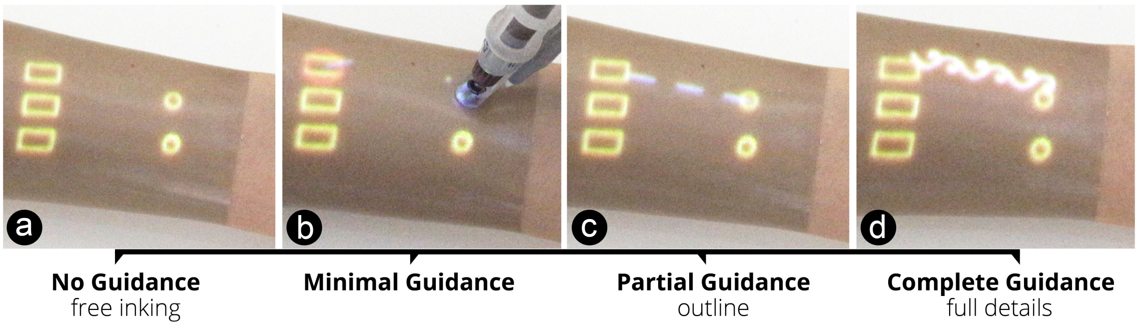


Figure 3.8 Varying levels of guidance to draw the conductive trace between two terminals. Without guidance, the user must draw freely and connect the correct pins (a). Minimal guidance highlights other pads the user can connect to (b). With partial guidance, the outline of the trace is displayed as a guide to route the connection (c). Full guidance shows a detailed pattern of the trace (d).

3.3.3 Inking Functional Conductive Traces

BodyStylus is designed to support inking electrically functional designs. In doing so, the user should be supported to create valid circuits while at the same time being free to creatively adjust the design to their desired visual aesthetics. For instance, instead of drawing the most efficient connection between two components, users might deliberately deviate to create a more aesthetically pleasing result or augment a trace with artistic elements, e.g., curlicues.

Again, the interaction methods are based on an interplay of *guidance*, which shows the users which traces or pads can connect, and *constraints*, which physically prevent the user from creating invalid connections (short circuits, wrong polarity – see also Figure 3.7). To achieve this, the system requires a model of the underlying electrical circuit; *BodyStylus* records the precise location of the pen on the body while *inking* and saves this data as a vector graphic. In addition, the system tracks the position and polarity of the embedded components and distinguishes between functional and artistic traces.

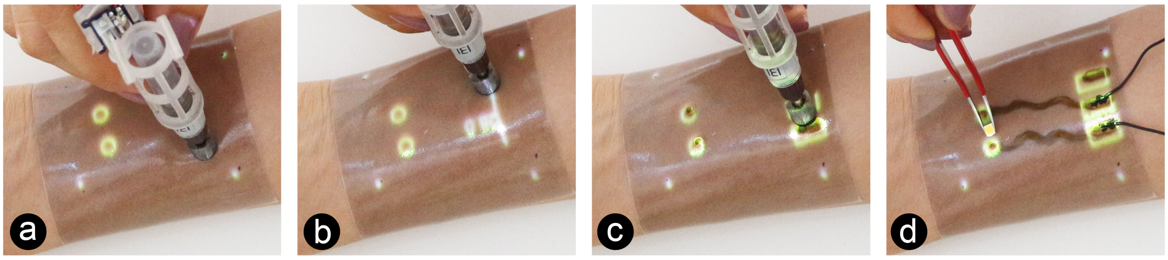


Figure 3.9 Placing electronic components. The user defines the location of a component's pad (a) and then drags the pen to define the distance and orientation of other pads (b). The system displays the footprints and the user starts inking (c). Finally, the user connects components using bare conductive (d).

For working with electrically functional traces, *BodyStylus* supports four levels of guidance (Figure 3.8). To work completely freely, users might choose *no guidance* (Figure 3.8a). To add some visual support *minimal guidance* can be chosen to highlight which other pads the current trace can connect to (Figure 3.8b). If using *partial guidance*, a direct path to a suggested destination pad is highlighted (Figure 3.8c, dashed line). Finally, *complete guidance* shows a path the user needs to trace to complete the circuit. This path might include basic routing (using the A* algorithm [74]) and can include visual styles for aesthetic effects (Figure 3.8d). The desired style is manually chosen from a library of patterns. The above guidance methods were described from left to right of the dimension space in Figure 3.5. They can be crossed with continuously varying levels of constraints, which change the margin of how close one may come to making an error before the pen retracts, moving up or down in Figure 3.5.

3.3.4 Placing Electronic Components

To assist users in adding standard electronic components to the epidermal circuits, *BodyStylus* contains a library of frequently used components. Our current implementation comprises resistors, capacitors, diodes, LEDs, inductors, and breakouts for connecting cables. Additional components can be implemented: for each component, the library stores the package type and function of pins.

Using this information, *BodyStylus* visually guides the user in placing the component. After selecting a component from the library, the user defines the location of the first pad (Figure 3.9a) and drags the pen to locate the last pad. The system guides the user by projecting the distance and angle between pads (Figure 3.9b). After releasing the pen, the system displays the component's footprints (Figure 3.9c). To make the design permanent, the user starts inking the pads. The system constraints painting outside the footprints' borders and connection between terminals by retracting the pen (Figure 3.9c). Finally,

Figure 3.10 Parametric components. The user places a bend sensor design on the wrist (a) and enhances its visual aesthetic with a curlicue pattern. She drags the center, which modifies the pattern on-the-fly (b). When moving the sensor to a body location the user cannot bend, the body constraint is highlighted in red (c).

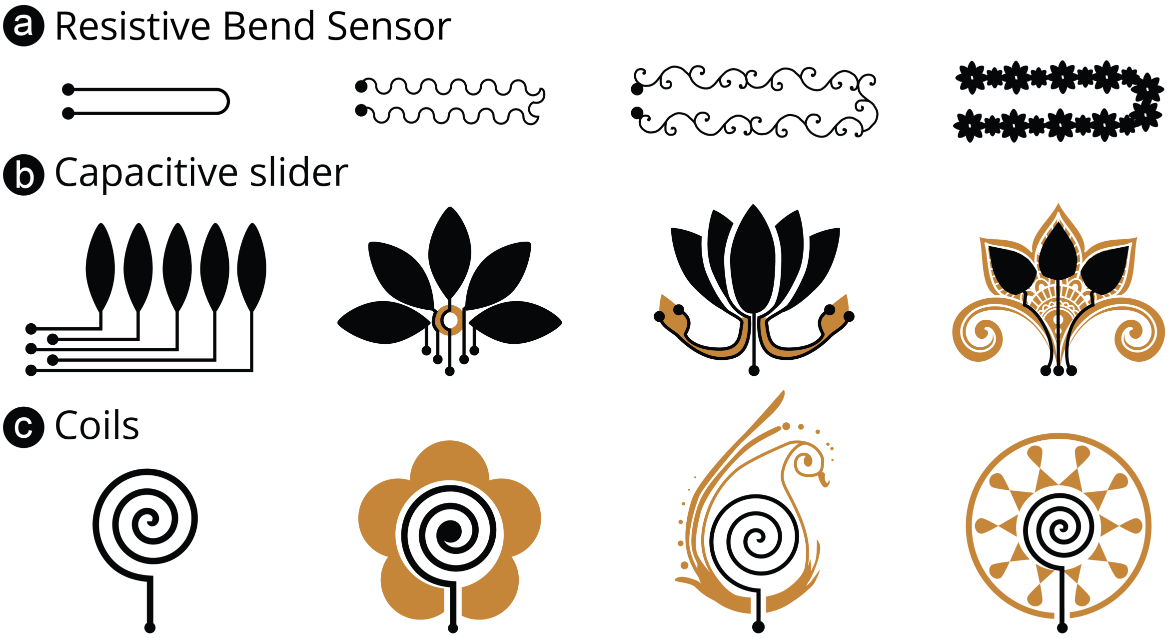


Figure 3.11 Custom free-form components can be created by selecting a parametric design (left-most column) and then customizing it in size and style.

the user places the components (e.g., LED) and connects the wires to the terminals using conductive paste as glue [30] (Figure 3.9d). Due to the requirements for strict constraints, these interactions are at the top of the dimension space in Figure 3.5.

3.3.5 Free-Form Components

Some commonly used types of electronic components can be hand-drawn in free-form using conductive ink, e.g., capacitive touch buttons, sliders, resistive bend sensors, strain gauges, or coils. These provide two opportunities: (a) the user can adjust their function to meet the specific demand of the system (number of windings in coil, length of bend sensor, number of pads on a slider, etc.), and (b) the user can adapt their aesthetics to match the overall design.

To provide support for these activities, *BodyStylus* offers a library of parametric designs for free-form components (resistive bend sensors, capacitive sliders, and coils; additional components can be added, similarly to electrical components). To create a free-form component, the user selects a parametric design and places it on the body (Figure 3.10a). The design can be moved, scaled, and adjusted in visual style (Figure 3.10b). If body placement constraints are violated (e.g., placing a bend sensor at a location that cannot bend), the component is highlighted in red color (Figure 3.10c). When satisfied, the user can ink the traces with conductive ink. Of note, the user is free to creatively deviate from the design and add ornaments that match the aesthetics of the overall design. Figure 3.11 shows various examples of how the user can flexibly adapt the basic parametric design to create different aesthetics. Of course, a specific aesthetic style can be encapsulated as part of a parametric design, too. For instance, the slider component allows the user to choose from several visual styles for the conductive line.

Creating parametric components borrows from previously introduced interaction methods: terminals are implemented as electrical components, connections within the free-from component are implemented as described for electrical traces, and specific shape features are implemented as described for aesthetic patterns. Of course, one can combine these with inked visual designs (see also Figure 3.9), and temporary sketching can be used for quickly testing ideas before permanently implementing them.

3.4 Validation

To validate the practical feasibility and to investigate opportunities and limitations of the *BodyStylus* concept, workflow, and example implementation, we present (a) examples of implemented designs alongside lessons learned, including a detailed walkthrough of how one of these designs was created and (b) feedback from a focus-group exploration with three-stage make-up artist trainees and a focus-group exploration with three engineers.

3.4.1 Application Cases Realized with *BodyStylus*

To demonstrate the end-to-end feasibility of epidermal interfaces, we present four example application cases we have designed and fabricated with *BodyStylus*: an interactive hand adornment, a wirelessly powered epidermal interface, a wristband control, and a decorative anklet. The functional prototypes are shown in Figure 4.20.

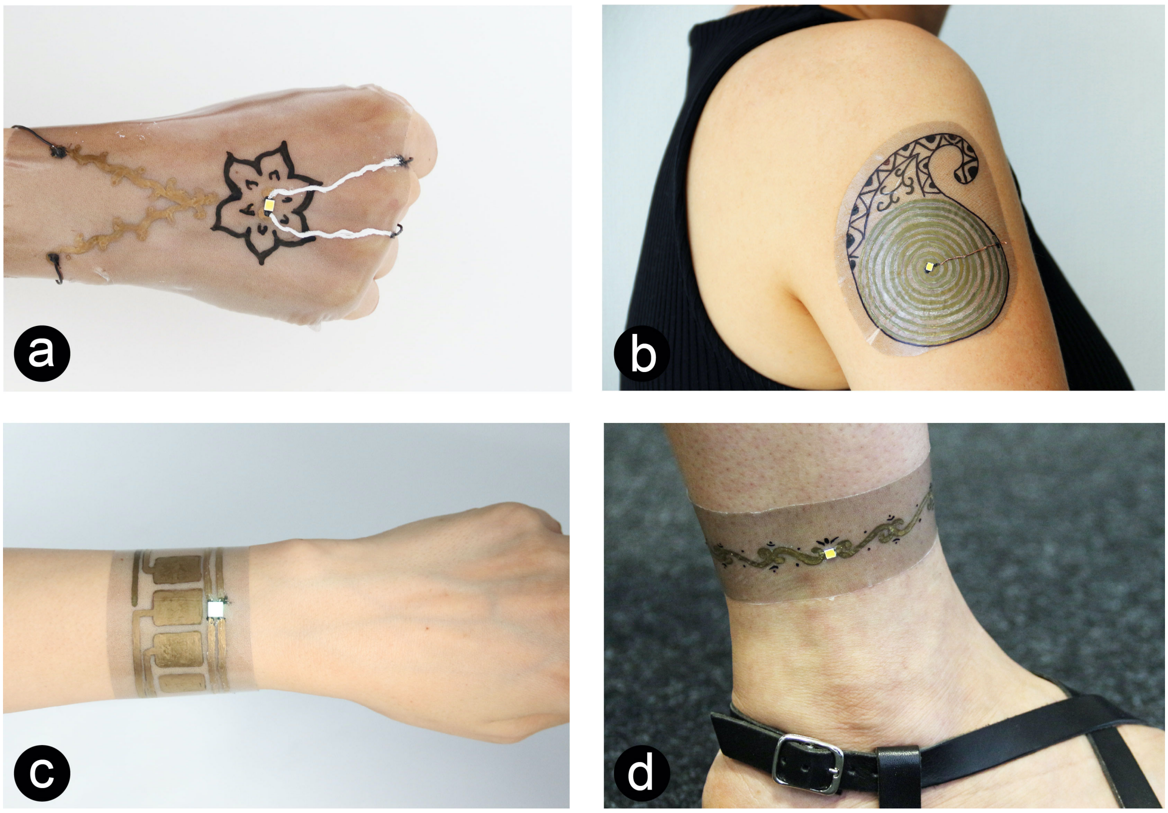


Figure 3.12 Interactive Application Cases. Interactive hand decoration that responds to body movement (a); wirelessly powered interface (b); wristband slider control (c); decorative anklet (d).

Interactive Hand Decoration

Inspired by interactive make-up such as Kinisi [222] and traditional bridal henna [149], we decided to create an interactive hand adornment that reacts to hand movement with light (Figure 4.20a). To describe the design and fabrication process, we present a step-by-step walk-through of this application case before presenting the three other application cases more briefly.

An important part of practicing body-arts is the preparation of the body, for example, a make-up artist applies a coat of primer to create a blank slate. Similarly, the empty substrate provides a blank slate for drawing on, and its placement already foreshadows its potential use. In this case, we placed the empty tattoo on the hand and wrist to support the implementation of a bend sensor and hand adornments.

Henna artists usually start their work by "placing dots" as guidance for subsequent patterns. We decided to start with the bend sensor and, similarly, initially placed control points that other parts of the circuit will later connect to. Once satisfied with the placement, the terminals were made permanent by inking them and choosing a visual style for the sensor (Figures 3.13a and 3.13b).

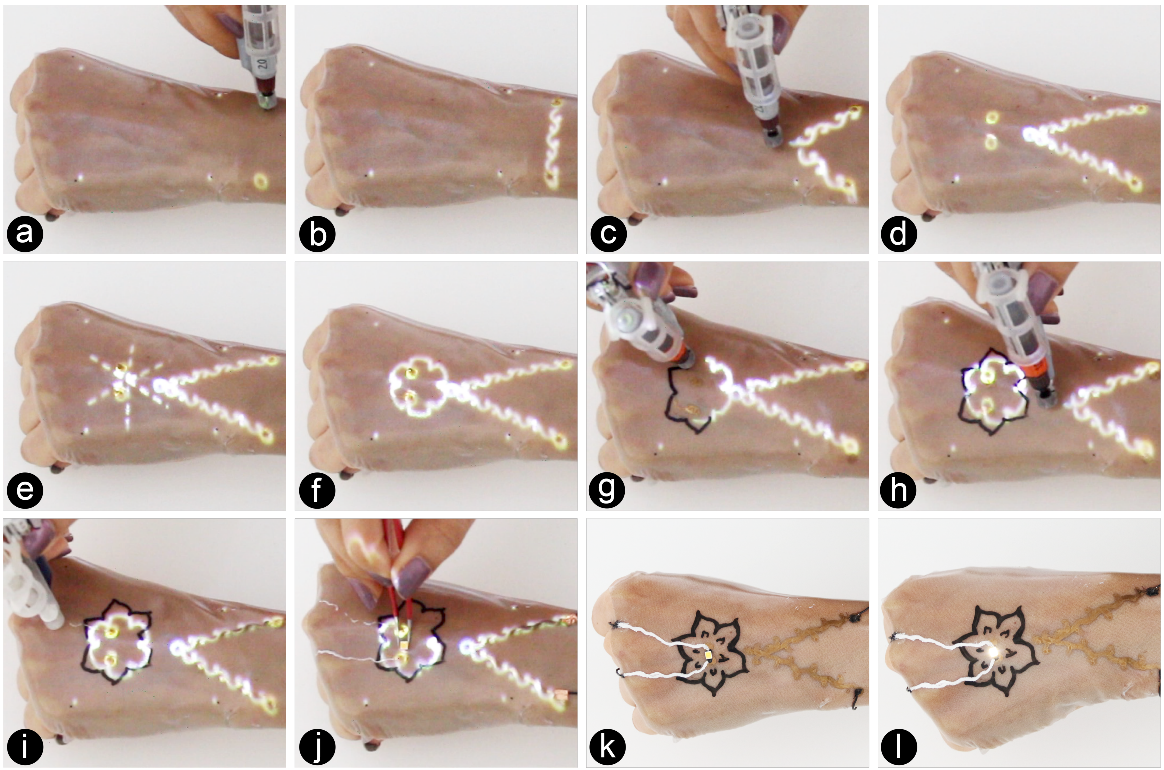


Figure 3.13 Design and fabrication steps of Shape Responsive Hand Decoration. *BodyStylus* supports the design and fabrication of visually aesthetic epidermal devices. The system guides the user by projecting the electronics, aesthetics, and body constraints and prevents logic error (constraints) by retracting the ink cartridge. The patterns are designed using conductive gold ink, silver pen, and regular black ink.

Artists consider the body's movements as an important reason for working on the body, to see the movement of the person and to see the response of the materials. Similarly, designing the bend sensor directly on the body allowed for exploring where exactly the wrist moves and deciding how the electrical traces cover this movement (Figure 3.13c). The position and size of the sensor relative to where movement occurs both influence the function of the design. At the same time, the pattern selected for the sensor shapes the design's aesthetics. For now, we did not ink the traces so we could return to tweak and fine-tune them later.

Next, we started sketching a flower design. As its center will be an LED, we first defined its footprint (Figure 3.13d). Then, we started sketching its adornments. Here the use of the radial symmetry tool was again reminiscent of traditional henna practice, e.g. drawing a crosshair as guidance. Similarly, we defined a 6-segment crosshair (Figure 3.13e), which we used to create a repeating pattern (Figure 3.13f). We then proceeded to immediately ink it using non-conductive ink (Figure 3.13g). Satisfied with our arrangement of the flower and sensor (Figure 3.13h), we also inked the sensor using gold ink as well as the connectors to the LED with conductive silver ink, using free inking without guidance (Figure 3.13i). In case of an error, *BodyStylus* would prevent us from creating a short circuit or from inadvertently connecting to the bend sensor.

In traditional henna practice, once bridal henna has dyed the skin and the paste is removed, it is not uncommon to add gemstones or other adornments, which are quite literally glued to the skin. Much in the same way, we finalized our design by placing the LED (Figure 3.13j) as well as connecting our prepared traces to an Arduino microcontroller. The finished piece can be seen in Figure 3.13k. When the wrist is flexed, the resistance in the bend sensor increases, and the LED is turned on (Figure 3.13l).

Wirelessly Powered Epidermal Interface

Inspired by [96], we set out to design and fabricate an epidermal interface for use on one's upper arm that is wirelessly powered (Figure 4.20b). It features an LED embedded inside an aesthetic line-art design. When powered using an off-the-shelf wireless charging coil [130] held at a close distance to the interface, the LED lights up. We selected the *Coil* free-form component and used visual guidance for creating a spiral that acts as a receiver coil for inductive power transmission. We used conductive yarn with a non-conductive coating to close the coil. As the conductivity of the conductive gold ink is not high enough for creating an induction coil, we realized the spiral with silver ink. For a visual style similar to henna dye, we then repeated the trace with gold ink. To make the design aesthetically pleasant, we experimented with several ornamental shapes and decided to surround the spiral with a paisley design, a key element of henna art, using free inking.

While creating the design, we realized that the use of impasto gel underneath the temporary tattoo substrate allows the substrate to be removed and re-applied multiple times. This creates opportunities for combining the benefits of on-body design with working off-body for tasks such as finishing complex patterns or debugging. The design has been done collaboratively on the body and fitted to the size and dimension of the user's arm, and then finalized on a table before it was transferred again to the body.

Wristband Slider Control

In our third application case, we aimed to investigate capacitive sensing. We have realized a wristband that comprises a capacitive slider with three segments and an RGB LED (Figure 4.20c). We selected the slider free-form component and used visual guidance to design and fit the size and number of segments to the wrist. The full constraint prevented us to connect the segments to each other inadvertently. Using projected guidance, we could design several sliders with different sizes, shapes, and numbers of segments at different angles to find the most suitable design without investing time to fabricate them. The device was created by a single user on her own body, and then connected and controlled with an Arduino microcontroller. In our proof-of-concept, the capacitive slider is used to adjust the color of the LED; in future implementations, a wireless data connection could be easily added to turn the device into a generic body-based controller, e.g. for controlling presentation slides, a music player, room lighting, amongst other options.

Decorative Anklet

To demonstrate that *BodyStylus* is applicable to various body locations, we realized a decorative anklet that features a blinking LED (Figure 4.20d). The LED footprint was selected from the component library and then placed in the desired position on the ankle. We used full detailed guidance to project the curlicue patterns between LED and the breakout footprint. The pattern was painted with gold ink and then connected to an SMD LED using bare conductive ink. An Arduino microcontroller is used to power and control the device.

3.4.2 Focus Group Explorations

We conducted two hands-on explorations to collect feedback of *BodyStylus* in use, following an approach that has proven valuable to connect with craft experts [23].

Method

The first exploration was conducted with three engineers familiar with designing circuitry, having backgrounds in embedded systems (E1, female, 24), robotics (E2, male, 28), and physiological sensing (E3, male, 30). The second group consisted of three professional stage make-up artists, all female, aged 26 (A1), 22 (A2), and 19 (A3), and experienced with a wide repertoire of on-body techniques, including the use of different mediums, pigments, and tools.

Before the exploration, we ensured that all participants had a working understanding of what epidermal devices are. After providing informed consent, both groups were asked to perform a series of tasks designed to elicit discussion and feedback. The sequence of tasks was arranged from simple to more complex, to allow participants to become familiar with *BodyStylus*. They were asked to (1) draw two intersecting lines on a mannequin, using the projection for guidance, and automatic retraction for constraints; (2) create a freehand pattern including one interface element (e.g., a button) on one of their colleagues; and (3) collaboratively design and implement a functional prototype. For the engineers, this was a generic controller, while the make-up artists were tasked to build a device for controlling RGB LEDs as used in a design they were working on at their theatre. We paused after each task, inviting critique and probing for observations and introspections. Sessions were audio-recorded, and photos of key situations were taken. Claims about the system, applications, or usage were extracted into individual files, reducing the 4 hours of recording to 33 minutes of tagged quotes. These quotes were thematically grouped as reported below.

Results & Discussion

Both groups were able to complete all tasks and provided valuable feedback. Most critically, after less than two hours of exploration, they were able to implement their own, functional, epidermal UI without intervention from our side. We now summarize the central findings by theme.

***BodyStylus* for Bodily Exploration.** Working on other people came naturally to the make-up artists, who do so in their day-to-day practice, while the engineers tended to show short moments of hesitation before inking each other. However, the engineers took advantage of working on the body rather than in CAD software, for example, to quickly test the positioning of UI elements on moving body parts (Figure 3.14b).

For task two we placed a tattoo substrate on the knuckles to see how participants might integrate these complex shapes into their designs. After asking participants to create input elements, we observed different ways in which designs reacted to body features: E1 lined up her interface elements with the knuckles, while E2 superimposed a button on a knuckle. E3 chose a different approach: He drew a strain gauge over the knuckles. Make-up artists engaged with the body differently for their designs. Instead of physiology, they focused primarily on the visibility of body parts. In the first design they created for task 2, A3 placed the button as far to the edge of the substrate as possible, with the intent of hiding the interface. Latter designs by A1 and A2 explored opportunities of using the button as an element visible to others.

Task three showed that the real-world application of the make-up artists was valuable in critically reflecting on design. While the engineers merely placed buttons on the arm (see Figure 3.14c), the make-up artists spent more time discussing and planning their final interface. They highlighted aspects such as the benefits of uni-manual vs. bi-manual operation, discussed which surfaces were better suited for pressure and which for touch input, and planned how to avoid unintentional activation. Working on the body led the

Figure 3.14 Engineers during focus group exploration. Inking each other (a), testing imagined UI elements (b), using guidance for implementing generic touch-controller (c).

Figure 3.15 Artists during focus group exploration. Inking on each other (a), applying substrate for functional prototype interface (b), completed epidermal interface controlling an RGB LED (c).

make-up artists not only to think about the visual layout of the interface but also to reflect on how the epidermal UI will interact with the body when in use.

***BodyStylus* as Artistic Hand Tool.** The engineers immediately praised the high ink-flow rate, which they felt made the device easy to use and promised robust circuits. Interestingly the make-up artists remarked negatively about the high ink-flow rate, especially with the fact that the ink-flow rate could not be continuously adjusted while inking. They felt that this inhibited their ability to produce delicate patterns and dynamic stroke thickness. While all participants were able to adapt to the offset between the inking tip and tracking tip, the make-up artists repeatedly stressed that it required active attention – especially on the uneven and compliant surfaces of the body. Interestingly this too was less of an issue for the engineers.

While we anticipated that participants might be apprehensive about working with an actuated tool that could retract based on predetermined rules, this was largely not the case. In addition, we did not observe any effect on the user's drawing while the pen was retracting. E1 mentioned that she felt the pen should be designed to minimize recoil when the inking tip was retracted. A2 explained that because she understands the behavior of *BodyStylus*, it is just a tool, much like the various pens and brushes she uses in her everyday practice.

***BodyStylus* as Error Prevention.** The engineers commented on the usefulness of constraints for positioning ICs and SMD components. Beyond that, E2 and E3 were especially excited that constraints provide them with the ability to draw finely detailed components such as strain gauges. E2 felt that such constraints might be a useful proxy for physiological sensing, for example in place of EMG, while E3 stated that he would enjoy inking with constraints on other surfaces than the body: *"whenever I build something with elastic materials and want to measure deformation, I'd love to use this for drawing sensors"*. For the make-up artists, the use of constraints for error prevention was mostly appreciated in the context of tasks requiring precise positioning and exact stroke lengths: *"oh, I want this next time I do eyes"* (A3). Nevertheless, it was also noted that constraints were only applicable to the very specific context of their craft, as in make-up one usually works with smooth gradients. Participants also expressed worry that over-reliance on the system might risk losing their manual skills. Along these lines, A3 stressed the importance of being able to override *constraints*.

***BodyStylus* as Assisting Guide.** With the simple examples created during the exploration, the engineers felt that they did not need the guidance features. They did speculate that it would be useful for someone with less knowledge of electronics. The make-up artists in fact did appreciate the guidance. For instance, A3 noted that the projected guides would blend in with make-up techniques making use of the human face's symmetry: *"there are these lines" [she points from nostril to outer edge of eye-socket] "which we use, and if you could project them, I would immediately use that"*. The guides blend in with their existing craft. For example, A1 mentioned that *"we already sometimes draw dots to guide the design"* and *"having this would be so much better"*. All three artists stated they would want to use guidance in their everyday practice.

***BodyStylus* Integrated in Day-to-Day Practice.** Interestingly, the artists' take on how the system should be used changed while they gained more experience with it. Initially, they envisioned making use of *BodyStylus* for final touches or for implementing a carefully thought-out design. Towards the end of the session, they noted that *BodyStylus* would be most handy as a sketching tool early on in the design process. They outlined how it could enable them to rapidly create a rough on-body draft so that they would have more time to work on the final details. The engineers questioned why the system was constrained to working on the body and highlighted that they would appreciate this approach for rapid prototyping in other contexts also.

Summary

Participants took advantage of working directly on the body by exploring the bodyscape, using the morphology of the body, and testing UI elements in situ. Differences in expectations and goals led to conflicting assessments of the physical pen prototype. In future iterations, dynamic ink flow should be considered, and the inking and sensing pen tips superimposed. Both engineers and artists appreciated constraints; however, constraints are more useful to engineers, who suggested applications for constraints beyond those we anticipated. Artists worry that overrelying on constraints might prevent maintaining and developing manual skills. Inversely, guidance appeared more suitable to the artists, who this time were the ones to suggest new applications. Finally, *BodyStylus* not only supported complete novices in the design and fabrication of a working prototype but also facilitated thinking about interaction and design.

3.5 Discussion

Occlusion and Calibration. Our implementation of in-situ visual feedback is currently restricted to a projector, which can cause occlusion on the skin while the user is manipulating the pen. Accurate projection mapping on a moving and deforming body is still an active research field. We implemented a simple calibration step for mapping the projection to the skin site's position, which however does not continuously update, hence requiring the skin site to remain immobile during the interaction. We experienced that this is acceptable when designing on limbs, as they can be placed in a comfortable resting position, but can be more challenging when designing on the torso. Note that thanks to the locally stable Anoto pattern, the position tracking remains accurate in case a skin site is moving in 3D space, which is important to maintain a correct digital model and ensure precise computer-controlled inking behavior. In future work, we plan to experiment with a pico projector built into the pen [200] and a head-mounted display for improving mobility and avoiding occlusions and to implement a more advanced projection mapping scheme as suggested in [153].

Focus on Hand and Arm. According to [71], the hand/arm space is the one most extensively used for on-body interfaces. Although our main focus is on these locations, evidence from the implemented example cases confirms that our approach works well on the demonstrated body parts (e.g., hand, forearm, upper arm, wrist, and ankle). It can be extended to other body parts, such as legs, thigh, and back (the latter requiring co-creation since self-inking is not possible on the back). On very challenging skin geometries, e.g., strongly convex body parts, the smart pen sometimes loses data points; interpolating the captured data allows estimating the pen position.

Substrate. Our current setup uses a substrate printed with dot patterns to enable the smart pen's position tracking. The choice of the substrate is a pragmatic choice to support this particular proof-of-concept implementation with precise tracking and an insulation layer. However, future iterations based on *BodyStylus* might be designed to operate more directly on the skin. One option is to make the substrate fully transparent, e.g., by printing the Anoto pattern using transparent IR-absorbing ink. Another option is to remove the substrate altogether, using a different position tracking technology, e.g., optical motion capturing, combined with an insulated spray primer. In addition, the strengths of *BodyStylus* include its support of skill-building activities using guidance and constraints as well as its ability to be used with self-sintering, skin-safe gold ink. As a result, artists could practice designs on themselves using guidance and constraints to commit drawn shapes to muscle memory, and then, removing the substrate and using only the pen with gold ink to recall the practiced shape when designing for a wearer.

Single-layer Substrate. While our approach is currently restricted to single-layer, conductive bridges can be realized with electrically insulated conductive yarn.

Modifying Traces. Lastly, while our system supports users before and during inking, it does not allow the user to modify traces after they have been made. Future work should investigate erasing interactions [146] to edit the design after inking.

3.6 Conclusion

Until now, design and fabrication processes for epidermal devices were indirect and detached from the human bodyscape. In this chapter, we presented a novel interactive approach that overcomes this limitation and allows us to design and craft epidermal devices directly on the body using a hand-held fabrication tool and self-sintering conductive ink. We present a palette of on-body design and fabrication techniques that leverage in-situ projected guidance and physical constraints to facilitate the creation of functional and aesthetic epidermal devices. *BodyStylus* is conceptually grounded on design considerations integrating circuit logic, aesthetic principles, body shape, and functionality, and expands upon the existing traditions of body markings by facilitating skill-building activities. We exemplify these parallels and starting points using four concrete application cases, where we further demonstrate how *BodyStylus* succeeds in creating functional epidermal devices. Our results show that our approach was positively received by stage make-up artists and engineers with an electronics background, and succeeded in moving the design and fabrication process from the computer and workbench onto the body, making it more direct and immediate.

The following chapter delves into an examination of a physical sketching technique aimed at seamlessly integrating high-resolution electrical interfaces into real-world objects while retaining their inherent properties.

4 Print-A-Sketch: Handheld Fabrication on Rich Materials

The emergence of digital fabrication has brought significant advancements in the design and development of interactive objects that are embedded in our environment. In this context, sketching in hardware (e.g.: [125; 170]) has become an increasingly important approach for creating and prototyping these interactive devices. Sketching in hardware captures not only the look and feel of such an object but also how one might interact with it [77]. If this hardware sketch is created directly on the object where it will be used, the effects of the object's affordances and physical environment on the interaction can also be experienced. Like physical sketching with pen and paper, situated sketching in hardware is usually a practice based on physical craft and skill, such as paperwork [170], textile crafts [66; 91; 105], or body art [162; 222].

Manual methods support expressive and creative design (e.g.: [15; 53]). However, they come with limitations in terms of both fabrication speed and precision: manual processes are poorly suited for interfacing with the miniature world of discrete components and embedded circuits. Printing, on the other hand, can be used to create complex [148] and high-resolution [60] electronics, and do so rapidly [99]. However, printing lacks the expressive and exploratory nature of hand-held sketching methods.

Combining desirable properties from both approaches, this chapter presents *Print-A-Sketch*[1], a handheld digital printer for rapid free-hand sketching of high-resolution electrical interfaces. *Print-A-Sketch* is designed to support free-hand sketching practices while providing the high resolution and detail of a piezo-inkjet printhead for added

[1]This chapter is based on a publication at CHI'22 [163]. As the first author, I led the development of the sketching technique and handheld tool, designed and fabricated the application cases, and conducted the evaluations.

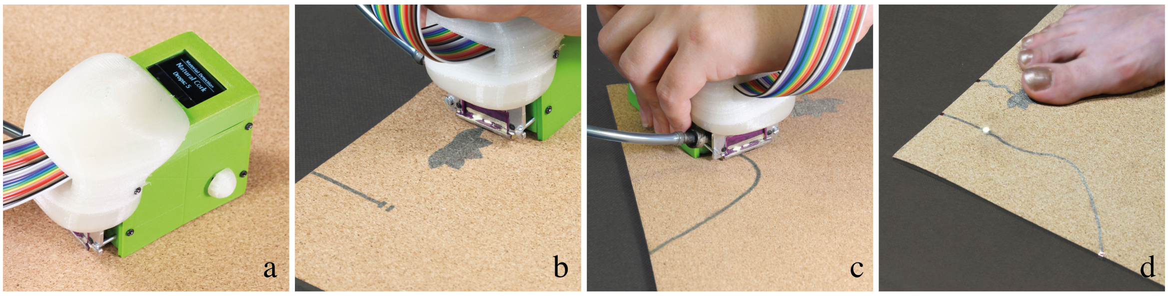


Figure 4.1 *Print-A-Sketch* uses a handheld printer (a) to create high-resolution conductive traces and circuits on everyday materials (b). It integrates the expressiveness of sketching with computational support (c), for direct fabrication of interactive interfaces on everyday surfaces and objects, such as this yoga mat (d).

precision and for interfacing with existing electrical components. In addition to the ability to print conductive traces, *Print-A-Sketch* is also augmented with a pair of visual sensors for sensing the print context. This information can be used to continuously adjust the printing parameters to free-hand motion, pre-existing sketch elements, as well as the material properties of the target material.

This chapter makes three primary contributions:

1) Conductive handheld printing: We present the first handheld printer for in-situ design and fabrication of customized circuits and sensors on existing objects. We demonstrate a) how to print high-resolution conductive structures with a readily available inkjet printhead on a wide array of materials and b) how to control the printhead in real-time, including detailed print parameters, using a commodity Arduino microcontroller platform. By open-sourcing the printer controller (hardware schematic and firmware for controlling the printhead), we envision this work to also make a practical contribution to the maker community. Evaluation results show good conductivity on many materials (e.g., various types of paper and cardboard, textiles, plywood, and stoneware) with sheet resistance ranging from 0.036 to 29.6 $\Omega/\square$.

2) Dynamic, context-aware handheld printing: We propose to integrate printing and sketching continuously. To this end, the handheld printer is context-aware: it continuously monitors how it is being moved on a surface, including its speed and relative position, using an optical motion sensor; furthermore, it detects patterns printed earlier with a wide-angle miniature camera. This contextual information allows the system to dynamically adapt print patterns in real-time with <0.5 mm precision. This opens up a novel hybrid design space where sketching and printing unify in one integrated task. We provide a conceptual overview of solution strategies and basic operating principles and present a working implementation.

3) Interaction techniques for on-the-fly design and fabrication of circuits and sensors: We explore the novel opportunities of interactions that integrate manual sketching with dynamic high-resolution printing. *Print-A-Sketch* offers techniques for printing different types of traces, such as serpentine conductors or footprints, for creating functional electronic components, and for adding high-resolution shapes and patterns. By detecting and dynamically adapting to prior printed patterns, the system can automatically connect to existing traces or pins, avoid short-circuits, route around obstacles, but also scan & print existing elements. Built-in measurement with 98% accuracy helps the user to print elements of defined length, parallel lines, and precise angles. Finally, integrating with existing components is nearly seamless, as optical sensing enables generating of IC footprints on-the-fly.

In this chapter, after discussing related work, we first present an overview of the three main challenges of free-hand sketching with a handheld printer. These relate to the irregular nature of human motion, the iterative nature of sketching, and interactions between print quality and various materials. After presenting the implementation we discuss each challenge separately. We then conclude with example applications and demonstrations of *Print-A-Sketch* in use.

4.1 Sketching with a Handheld Printer

An artist might sketch a painting with a pencil, switch to a large soft brush for painting the general structure, switch back to a tool with a finer tip for details, or may use a ruler or mask to create the desired pattern. For this artist, the ultimate tool might be one that changes the properties of its tip or selectively dispenses paint in a desired pattern, as required by the task at hand. Such a dynamic tool would also be invaluable for sketching in hardware. It would allow the designer to draw thick leads for connecting power traces, thin lines for connecting control signals to fine-pitched integrated circuits (ICs), as well as decorative lines and custom patterns. With *Print-A-Sketch* we demonstrate that a commercial printhead can be used to implement such a dynamic tool. Integrated into a handheld tool, the printhead can be used for sketching electrical interfaces on the objects and surfaces around us, supporting the expressivity of human motion, while providing the precision and detail required for interfacing with commercial electronics.

However, a conventional printhead operates in a relatively controlled environment. Printhead designs typically make a number of assumptions. For example, printers dispense ink on a controlled surface – typically a sheet of paper. The image to be printed is not modified once the print has started. The printhead moves with a known, fixed, speed.

All aspects of using a printhead for free-hand sketching on everyday objects break these assumptions.

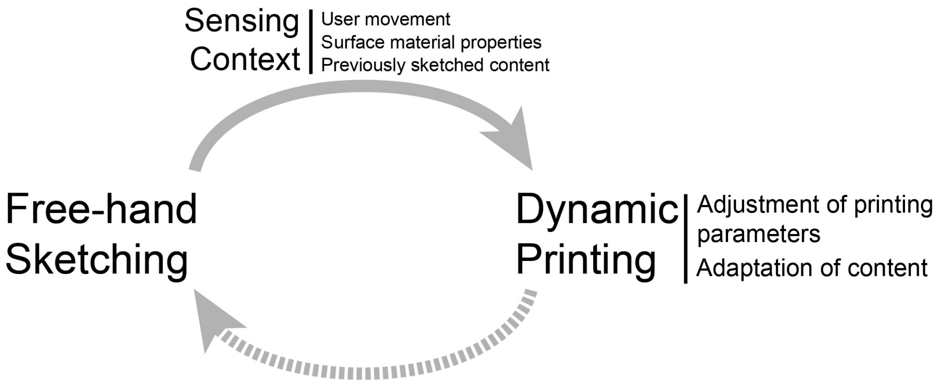


Figure 4.2 Free-hand sketching and dynamic printing are intertwined: Sketching is done by printing, and printing requires sketching. This tight coupling of printing and sketching requires continuous sensing of hand movements to ensure consistent quality of resulting prints. While adapting to free-hand movement, other information such as previously sketched traces or substrate material is used to contextually adapt the print.

Therefore, this vision creates a number of challenges. (1) **Free-hand** sketching comes with hand movement. Our hands move at varying speeds, they do not always move in straight lines, and they might shake or shiver. This is a problem not only for printing small high-detail features, such as IC footprints but also for ensuring constant and continuous dispensing of ink. (2) **Sketching is incremental and iterative** and needs to support spontaneous decision-making, but printing iteratively can easily lead to misaligned sections or additions which break existing functionality. Finally, (3) **everyday objects and surface** come in many materials; the amount of ink required for a conductive trace can change from material to material.

To address these challenges we suggest that a handheld printing device needs to be paired with appropriate sensing capabilities. This can then support dynamic control, where the printing process is continuously adjusted in real-time to contextual factors (Figure 4.2). Relevant context includes a) the current position of the device and its movement, to compensate for the variations of hand movement, b) existing patterns on a surface, so designs can be modified on-the-fly to accommodate any previously sketched content, and c) the material properties of the surface, such that ink dispensing can be adjusted for high-quality results. We do so by the example of *Print-A-Sketch*, a prototype handheld printer consisting of a Xaar piezoelectric printhead with 128 nozzles, which are

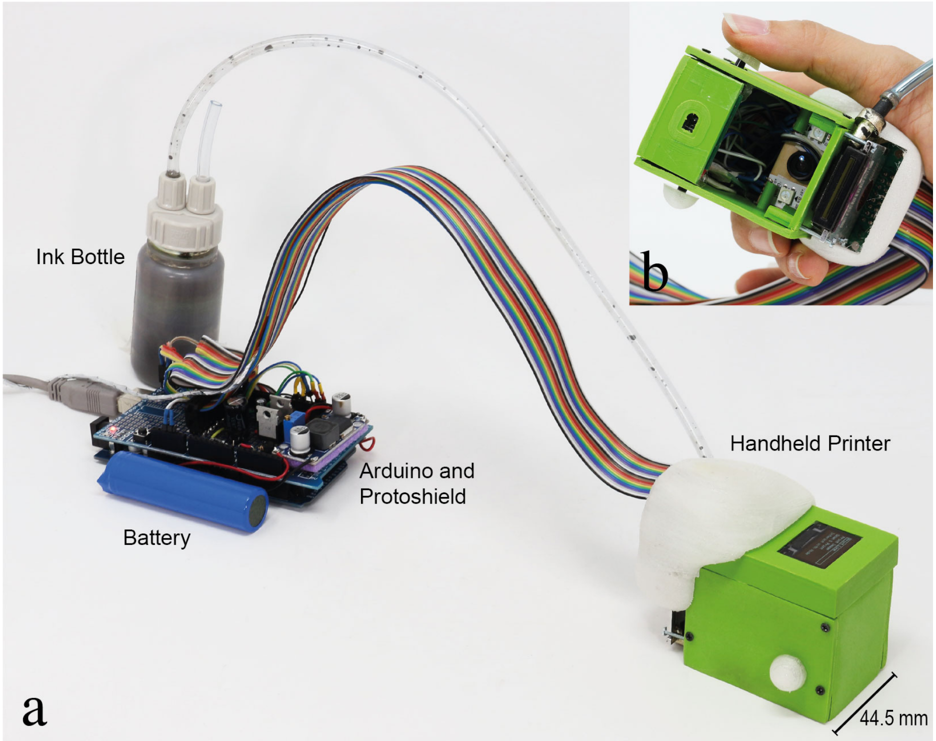


Figure 4.3 (a) The hardware setup includes the handheld printer, conductive ink, Arduino Mega, and Protoshield. (b) Bottom view, revealing integrated optical flow sensor and RGB camera.

complemented by two optical sensors for detecting movement, materials and previously sketched designs (Figure 4.3).

In the next section, we present the implementation of *Print-A-Sketch* and detail how the printhead is paired with sensing technology. Then we discuss how *Print-A-Sketch* addresses the three main challenges and present novel ways of sketching in hardware.

4.2 Implementation

Print-A-Sketch is designed around a flexible, hackable, high-resolution inkjet printhead, paired with two visual sensors capable of detecting user movement, surface visual texture, and material properties (Figure 4.3). Here we first discuss conductive printing, then sensing, and finally the user interface for controlling *Print-A-Sketch*.

4.2.1 Conductive Printing in a Handheld Form Factor

Piezoelectric Inkjet Printhead. *Print-A-Sketch* makes use of *piezoelectric* inkjet technology for printing conductive circuits. While there is a range of ready-made compact, handheld *thermal* inkjet printers (e.g., Prinker [166], COLOP e-mark [29]) commercially available, they have undesirable effects on conductive inks. This issue is caused by their operating principle where the ink is heated, up to the point where it vaporizes and expands out of the nozzle. Heating functional materials (e.g., silver nano-particle ink) may lead to degeneration and loss of the desired functionality, which is a known issue in Material Science research [32; 99]. In contrast, piezoelectric printheads create patterns of expansion and contraction to jet the ink out of the nozzle in response to an electric impulse. Hence, they can work with a wider range of inks such as electro-conductive ink [46]. For this reason, we decided on a custom setup featuring a Xaar 128 piezoelectric printhead (Xaar plc, approx. 220-240$). With its 16.5 mm printhead, Xaar 128 has a conveniently small form factor (37.2 x 11.3 x 40.8 mm). It is lightweight (15.5 g) and able to function in various orientations, which makes it suitable for printing on diverse surfaces and geometries. In addition, it allows for high-resolution prints (200 x 200 dpi); an important factor for printing cohesive, functional circuits. We use the integrated control circuitry of the printhead for low-level adjustment and tuning of the printing parameters [158][2].

Driver Board and Firmware. All electrical components are assembled on an Arduino Protoshield to ease replication (Figure 4.3). The Protoshield provides a high voltage (35 V) to power the printhead using a rechargeable battery (3.7V, 2200mAh), and links up the printhead with an Arduino. The Arduino also controls the power up and power down sequencing (cf., Tables 6.3 and 6.4 in [158]) which is essential for the printhead to function. The custom firmware is written in C++ and deployed on an Arduino Mega. It controls low-level parameters of the printhead and samples the data from sensors. Print data and nozzle firing parameters are updated based on sensor data and sent to the printhead using SPI. The circuit diagram of our hardware setup and firmware code for controlling the printhead is made available as open-source.[3]

Conductive Ink. Selecting functional inks that are suitable to be printed with a *piezoelectric inkjet printhead* is challenging because the ink's composition and viscosity need to be compatible with the printhead (for our model below 12 cps). Simultaneously, the inks should be highly conductive and applicable to surfaces that are both porous and non-porous. In addition, printing on arbitrary everyday surfaces such as wood, textiles, or ceramic tiles requires a relatively low curing temperature to enable in-situ curing, for instance with a household iron (max. 100°C), a blow-dryer, or a heat lamp. From the available off-the-shelf conductive inks, we selected nanosilver ink (Metalon® JS-A102A) from

[2]The printhead we use is active-low. For simplicity and generalization, our figures show active-high control

[3]https://hci.cs.uni-saarland.de/projects/print-a-sketch/

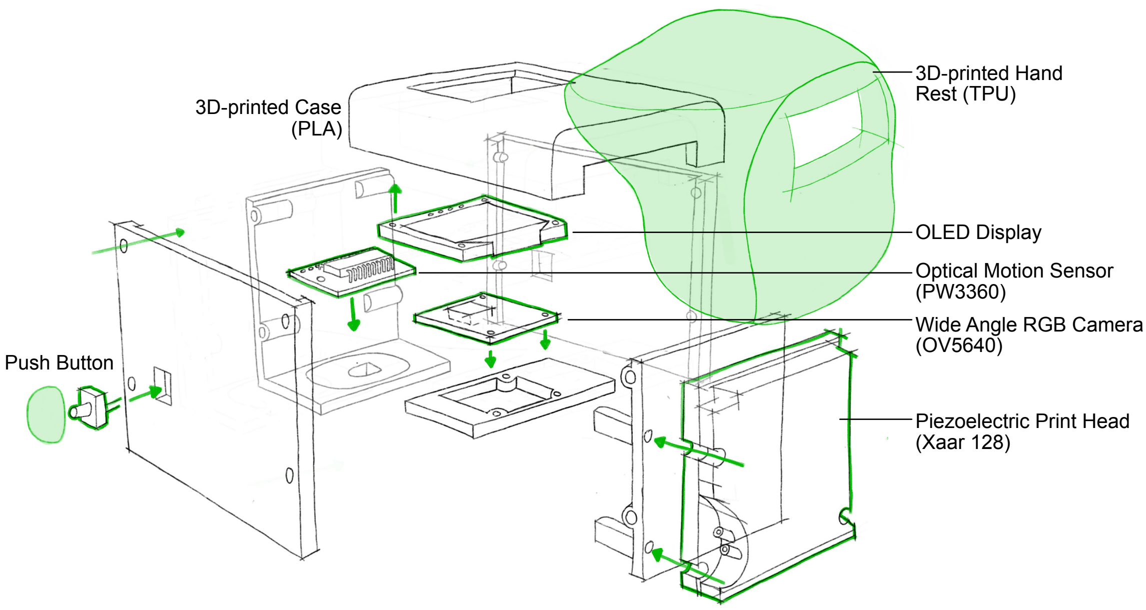


Figure 4.4 The handheld printer includes a printhead, an optical flow sensor, and an RGB camera for sensing and tracking. A display and two buttons enable the selection functions of the device.

Novacentrix [142] which is highly conductive (0.100 Ω/□ on Melinex ST505) and offers the needed viscosity. The 50 ml ink container is connected to the printhead using a 3 mm ID tube. In the current implementation power circuitry and ink container are offloaded from the hand-held printer. Overall, the handheld printer (green box in Figure 4.3) and the complete hardware setup (Arduino, battery, and ink bottle) weigh 110 g and 200 g, respectively.

4.2.2 Movement and Context Sensing

Print-A-Sketch draws strength from continuously adapting to the surface context and adjusting the print parameters accordingly. To capture *device movement* and the surface's *visual texture*, we employ a combination of two different optical sensors, an RGB camera and an infrared-based optical flow camera (Figure 5.11).

Optical Flow Camera. The optical flow sensor captures precise records of the device's movement speed, movement direction, relative orientation, and hand jitter. This sensor's accuracy is crucial for the quality of the resulting print. We thus opted for a highly accurate

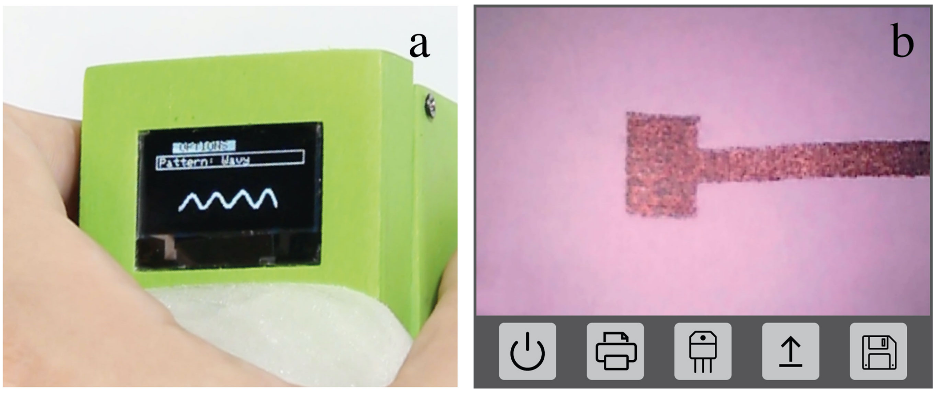


Figure 4.5 *Print-A-Sketch* user interface: (a) the user controls most functionality directly on the hand-held tool (here: selecting the trace pattern), (b) the backend interface supports additional functionality (e.g., showing the live camera view).

and fast sensor, namely PMW3360 [157], which is commonly used in optical gaming mice. It features a resolution of up to 12,000 Counts Per Inch (CPI), a speed of 250 inches per second, and an acceleration of 50 g. The sensor measures displacement by capturing the surface image and calculating the movement direction and speed. The data is then passed on via the SPI interface to the microcontroller, where it serves to adjust the printhead's firing frequency, and printing direction, and to determine which of the 128 nozzles need to be fired. We further use the lift detection feature of the sensor to identify hovering states and signal the printhead accordingly (start/stop sequences).

RGB Camera. The presence or absence of a previously printed trace is a change to the visual texture of the surface. To detect visual textures including previously printed traces, we use a 2592 × 1944 pixel camera module (OV5640) [214] which is connected to a laptop via USB. We installed the camera inside the handheld tool's case pointing down toward the print surface, together with two LEDs for controlled illumination (Figure 4.3 b). Previously printed designs or other surface features are detected and tracked visually using OpenCV's blob detection feature. In addition, data from the RGB camera and optical flow sensor are combined for detecting material properties and adjusting the print properties accordingly.

4.2.3 User Interface

To support rapid and convenient sketching in hardware, users are provided with a two-part user interface. Most functionality can be triggered directly on the hand-held tool (e.g., selecting the trace pattern, confirming/rejecting to connect to the detected trace) (Figure 4.5 a). For this purpose, it features an OLED display module [85] on its top, and a push-button on each side of the case (Figure 5.11). The user's hand is supported through an ergonomic handrest printed from TPU. For more fine-grained control and access to less frequently used functionalities (e.g., uploading a new design, defining a new component) we implemented a backend interface in Python that runs on a Macbook Pro (2.3 GHz Core i5) (Figure 4.5 b). The backend interface shows the live camera view and also serves as a link between all of the components: it communicates with the Arduino microcontroller using a serial protocol to control the printhead and analyzes the visual data. When printing images, it sends linewise pixel data to the Arduino. Both the backend interface as well as the Arduino firmware can manipulate the data stream based on contextual information.

4.3 Adapting to Free-Hand Motion

In conventional desktop inkjet printers, the position and movement of the printhead are precisely controlled by stepper motors. In contrast, with the artist or draftsman freely moving their brush or pen, processes of painting or sketching are dynamic: speed and direction of the sketching tool are varied in one fluent motion. Preserving this freedom of movement in a handheld printer requires dynamically adapting the print motif to speed and direction of movement. In this section, we present solutions to ensure **consistent print dimensions and quality** independent of the speed of movement–an aspect that is particularly critical when printing conductors, as they need to be end-to-end conductive. We also show how to automatically compensate for unsteady motion or shaky hands, to realize **steady free-hand printing**.

4.3.1 Consistent Print Dimensions and Quality

With constant speed and line-wise movement, ink delivery in desktop printers is less complex than for handheld printing: the nozzles located on the printhead jet ink droplets at constant time intervals that are fixed relative to the printhead's speed. The *firing frequency* is constant. This allows for printing patterns with consistent dimensions and quality.

In contrast, the dynamic speed of movement of a handheld printhead affects the density of the delivered print. This causes two related problems: discontinuous traces for freehand sketching and distorted images of digital print. For freehand sketching, fast movement causes jetted droplets to spread out wider, creating discontinuous traces that are not end-to-end conductive. The slow movement, in contrast, causes droplets to encroach on one another, creating image bleed and possibly even short circuits. Figure 4.6 a) illustrates the effect. Figure 4.6 c) shows an example printed with varied movement speed, but the constant frequency of ink jetting, leading to a distorted and discontinuous print. Similarly, if one wishes to digitally print patterns, increasing movement speed elongates the pattern, while slow movement compresses it.

To address the issues cause by free-hand movement, *Print-A-Sketch* precisely adjusts the frequency of ink delivery (modulated by a waveform signal with a signal where changes in the level cause droplet release), to the speed of the printhead traveling over the substrate. This is illustrated in Figure 4.6 b). In our setup, the optical flow camera detects the distance and direction of movement, from which we calculate movement speed. It ensures that irrespective of movement speed, a pattern remains undistorted and printed with consistent dimensions while also maintaining consistent droplet density (Figure 4.6 d).

Print-A-Sketch does not have a set direction in which traces must be drawn. The user could reverse the direction of motion. When this happens the order of lines sent to the printhead via the SPI interface is reversed.

Accuracy of Measuring and Printing

To test the accuracy with which the printer can measure movement speed and adjust the print accordingly, we performed a technical experiment. It involved printing a 20 mm long trace at three different speeds.

A linear slider stepper motor [135] was used to move *Print-A-Sketch* at 6, 8, and 12 mm/s. At each speed, the trace was printed 3 times on standard office paper. Then the length of the printed traces was measured and the offset from the ideal length (20 mm) was calculated. The results show a high accuracy of 98.0% (Absolute mean error: 0.4 mm, SD: 0.2 mm), with a maximum offset of 0.73 mm at a speed of 12 mm/s. This also suggests that *Print-A-Sketch* might be used as a measuring device when designing on the fly.

4.3.2 Steady Free-Hand Printing

Free-hand sketching comes with all forms of unintentional hand motion. This introduces artifacts due to unsteady motion or shaky hands, which, in a sketch simply become a part of the resulting drawing. When, however, printing a detailed pattern, an IC footprint, or routing multiple traces, lateral movement by the hand holding the printer causes wobbly and jittery prints, or – in the worst case, if designing electronics – even creates undesired short circuits.

Print-A-Sketch continuously detects lateral movement of the print using the optical flow sensor and counteracts by adjusting the printed image on-the-fly. As illustrated in Figure 4.7, the print pattern is adjusted to side-ways motion by shifting the bits in the array

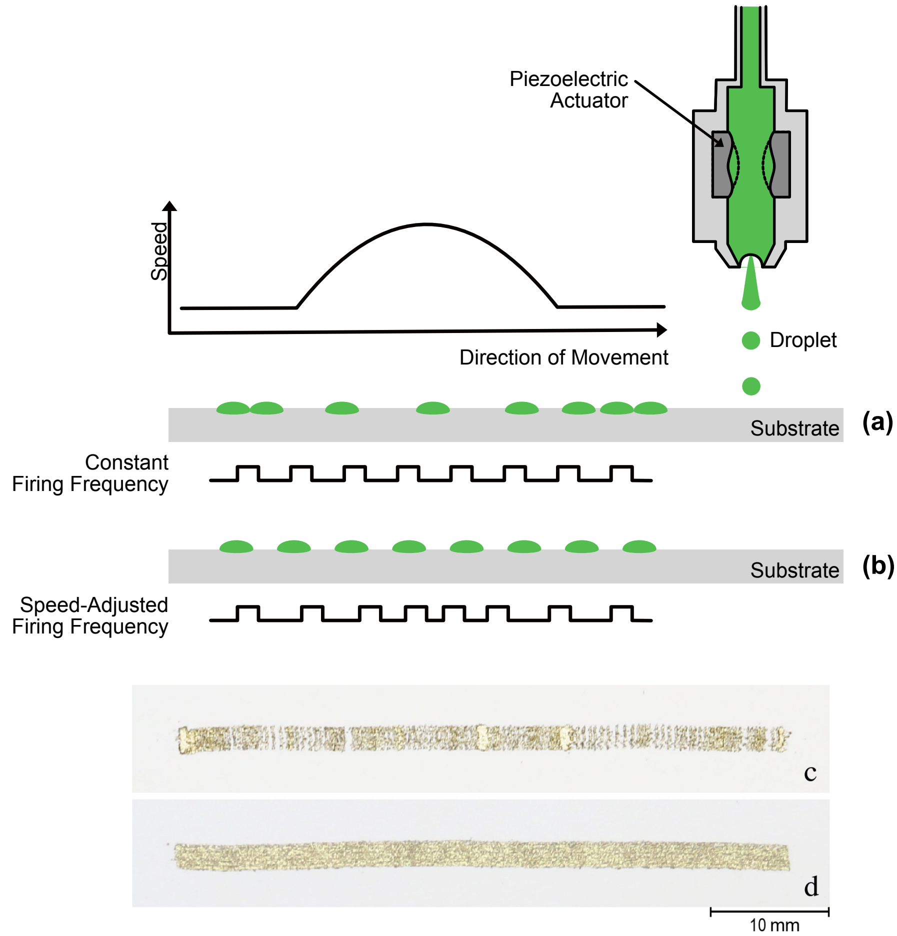


Figure 4.6 Top (side view): (a) Irregular free-hand movement causes irregular gaps between droplets when the firing frequency is constant; (b) Speed-adjusted firing frequency allows to maintain of a consistent resolution. **Bottom**: (c) Free-hand printed trace with constant firing frequency cannot be used as a conductor and also has visual artifacts (d) Free-hand printed trace with speed adjusted firing frequency is visually consistent and has good conductance

of data that is to be printed to the left or right, corresponding to the extent of side-ways motion that has happened since the last row of droplets was printed. This cancels out artifacts created by movement orthogonal to the printing direction, as long as the extent of movement minus the width of the printed pattern does not exceed the width of the nozzle array. In practice, assuming the trace is centered under the printhead, the possible deviation is $\pm(printheadWidth - traceWidth)/2$ (e.g.: ±6,25 mm for a 4 mm trace).

To demonstrate the performance of motion correction in practice, we printed several straight lines with movement correction activated. Figure 4.7 (middle) shows three different examples of printing with lateral movement correction. Figure 4.7 (bottom) shows an exemplary trace, it is 120 mm long, and 4 mm wide allowing correction for up to 6.25mm. The trace was printed on standard office paper while moving the handheld

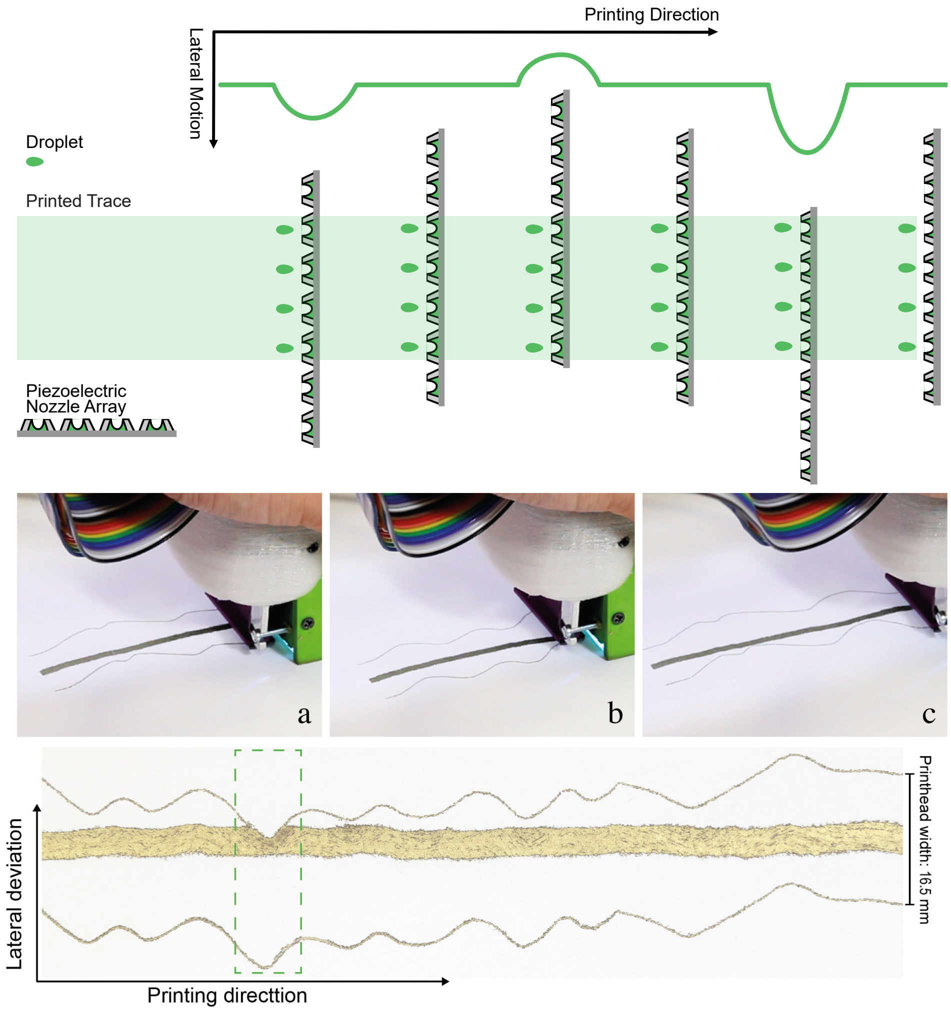


Figure 4.7 Top (viewed from the top): To print a straight line, even when the user introduces lateral motion, nozzles can be selectively used to counteract the moment. **Middle**: three examples of tracing with motion compensation. **Bottom**: Conductive trace printed with motion correction. The thin traces on top and bottom show the strong lateral movement of the print head while the trace was printed.

tool back and forth orthogonally to the printing direction. For reference, in Figure 4.7, while printing, the leftmost and rightmost nozzles of the printhead were continuously printing a thin trace; these traces visualize the lateral movement and indicate the printing range. For all other nozzles in-between, we applied the motion correction technique. As visible in Figure 4.7 (bottom row), despite the extreme lateral deviation of the printhead, the printed trace is steady. However, as highlighted in green, exceeding the width of the printhead causes artifacts. These errors, however, were caused intentionally, typically visual feedback from *Print-A-Sketch*'s display would enable the user to prevent creating deviation of this magnitude.

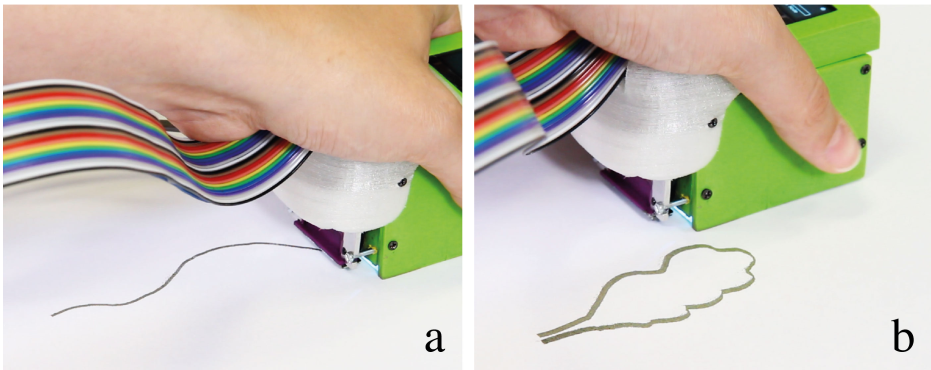


Figure 4.8 Shapes and curves created by free-hand motion: In both examples, the digital printhead produces a constant straight line, following the human motion.

Together, the ability to print consistent patterns and counteracting user movement form the basis for enabling the sketching of functional circuits as we outline in the next section.

4.4 Supporting Sketching

Like an artist who freely moves their brush over their canvas, users can freely move *Print-A-Sketch* to create ad-hoc free-hand patterns (see Figure 4.8). In this section, we present how such free sketching can be enhanced by high-resolution patterns that the handheld printhead is printing dynamically. As demonstrated in the following, this combination creates a new sketching experience that is characterized by a synergetic workflow of free sketching and printing.

In addition to adding **digital detail**, this enables contextually informed **adaptive printing**, where the printhead adds intelligently to the tracing motion of the user. Finally, the RGB camera together with a library of electronic components supports the user in easy **integration with existing electrical components**.

4.4.1 Adding Digital Detail

While users can manipulate the overall structure and design of the sketch by using *Print-A-Sketch* like a pen or a brush, the digital printhead of *Print-A-Sketch* can be used to further refine and manipulate details:

Custom Brushes and Line Styles

Both in physical and digital sketching, it is common to adjust the tool to the intended trace style. For example, in manual sketching one might select from different levels of graphite

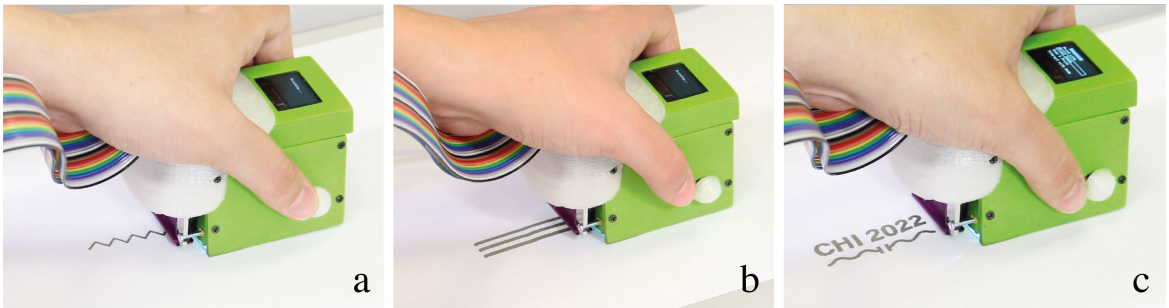


Figure 4.9 Custom Brushes: a) Printing traces with custom pattern and width, b) printing multiple lines, c) stamping an uploaded image.

hardness; with digital sketching tools, one might select a textured digital brush for the desired effect. Following this metaphor *Print-A-Sketch* enables the user to use 'brushes' with different characteristics. For example, varying trace widths, styles (e.g., solid, dashed, or dotted), and patterns (e.g., serpentine and zigzag) can be selected directly on the device (see Figure 4.9a). Printing multiple parallel lines is also possible; in this case, the user defines the number of traces and the distance between parallel lines (Figure 4.9b).

Stamping Shapes

Similarly to a traditional stamp, *Print-A-Sketch* can be used to create basic shapes (including circles or rectangles) for users to develop into more complex designs. These basic shapes can also be used for creating electrically functional designs. For example, circles can be used as capacitive buttons, or a series of *stamped* triangles can be used to form an interdigitated slider. Users can add custom shapes by uploading a bitmap image to the backend interface.

Scan & Print

Custom images and shapes can also be manually designed. For example, the desired shape can be drawn with a pen or pencil, and then scanned by placing and moving the hand-held printer over it. Subsequently, following a "copy and paste" metaphor, they can be re-printed at a different location. This technique is especially promising for building on or repairing circuit elements where no template exists (e.g., because they have been created free-hand) or where the template is unknown for another reason (Figure 4.10).

Figure 4.10 Scan & Print: a) Sketching custom design, b) scanning, c) and printing the design.

4.4.2 Adaptive Printing

We use an RGB camera to enable *Print-A-Sketch* to sense previously sketched traces and footprints. In the same way as *Print-A-Sketch* can detect free-hand motion and intelligently compensate for it, *Print-A-Sketch* can also intelligently react to existing sketches. The precise control of nozzles allowed us to implement several automation routines to support creating functional electronic sketches:

Stopping and Resuming Traces

Sketching includes pausing to review and rethink a sketch, or returning to previously sketched traces to add further details. To allow for the continuation of previously printed traces or to connect a new trace to a previously printed pad, a handheld printing device needs to adapt ink delivery to existing traces to ensure precise alignment of old and new prints. Compared to common sketching tools, e.g., pencils, this is challenging to do manually as the printhead's miniaturized array of nozzles does not resemble a single tip and does not offer sufficient cues for which nozzles will activate when delivering a given pattern.

To overcome this issue, contact of *Print-A-Sketch* with the surface is detected by the optical flow camera. Then monitoring of the surface's visual features, for instance, previously printed traces or pads for connecting components, enables continuing the print where one had left off.

When a user intends to resume a print, e.g., connecting a fresh trace to a previously printed trace or pad, they place the handheld printer roughly on the pre-existing element and move it in the desired direction (Figure 4.11). The image from the RGB camera is analyzed using blob detection. The software then autonomously determines the exact position of the connection point in the print area and modifies pixel data and data stream accordingly by a bit-wise shifting of the image data in the shift registers. As a result, only those nozzles that are spatially aligned with the connection point are activated (pixel

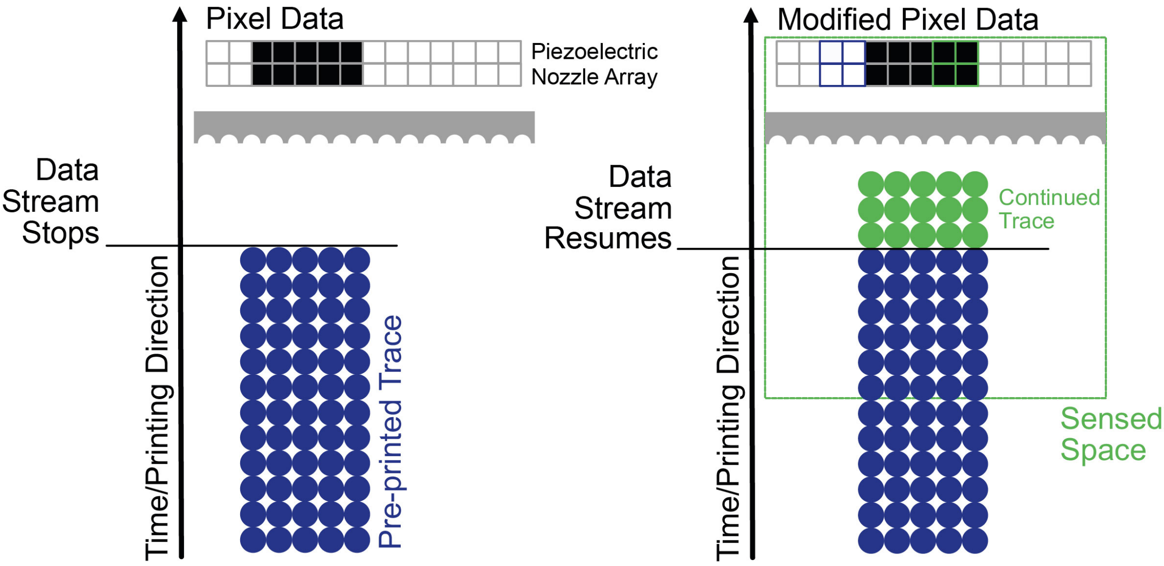


Figure 4.11 Left: Lifting the printer from the surface causes the data stream to stop and the nozzles to deactivate **Right**: To resume the print, the printhead is placed roughly over the pre-printed trace. The trace is detected using the RGB camera. If the printhead is not perfectly aligned, as seen here, the data is shifted to correct for the offset. Printing automatically resumes where the previous trace left off.

data, perpendicular to print direction), and fire at the precise point in time when the nozzle array traverses the pre-existing element's endpoint (data stream, aligned with print direction). We illustrate this principle in Figure 4.11.

Connect-to Objects

This principle is also used for stopping the print when connecting to a target component, as shown in Figure 4.12 top. When *Print-A-Sketch* encounters existing printed features, they are identified using blob-detection. *Print-A-Sketch* than provides the user with a *connect-to* function. If only one blob is detected, the system identifies the center of the blob's side facing the printhead as the point of connection. This generic implementation works for traces of diverse styles as well as pads or electrodes of rectangular or circular shape. If multiple blobs are visible in the camera view, the user has the option to select the desired element in the user interface.

Note that in frame c the trace dynamically adapted its direction to connect to the target pad. For starting a trace on an object, blob detection is used to decide when the print should begin (Figure 4.12, bottom).

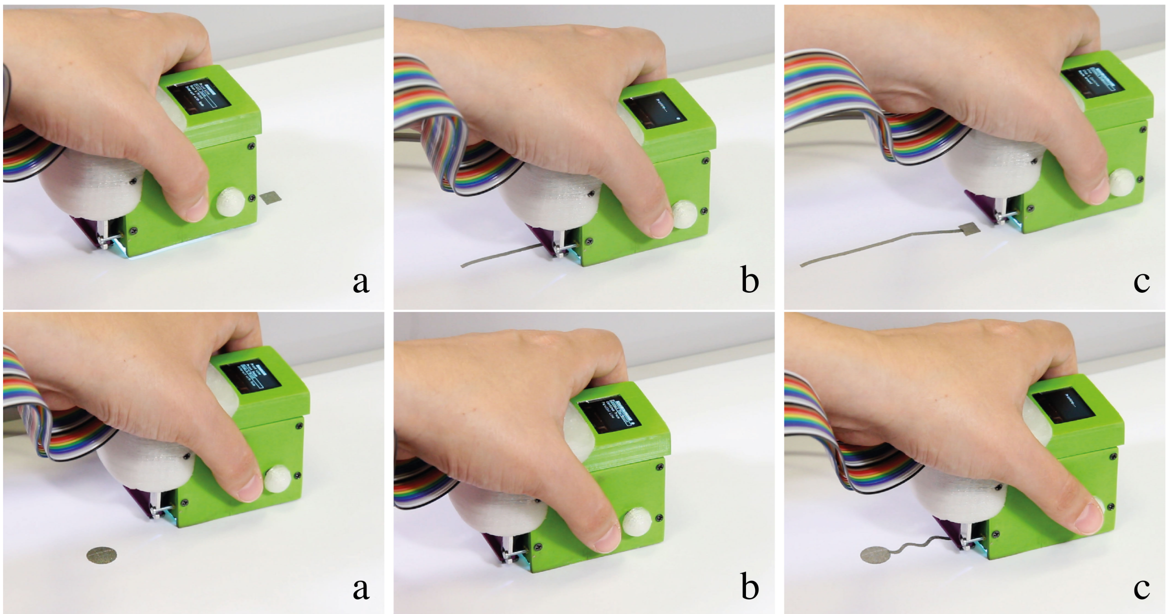


Figure 4.12 Top: Connecting to existing footprint: a) selecting the endpoint, b) handheld tool detects the footprint, c) printing trace is routed to the footprint. **Bottom**: Starting from existing footprint: a) selecting the start point, b) handheld tool detects the footprint, the user selects the width and pattern of the trace, c) printing the trace.

Routing

While intelligent stopping and starting of prints is a required utility for functional designs, *Print-A-Sketch* can go one step further and also proactively support users in creating electrical designs:

First, *Print-A-Sketch* enables printing corners with precise angles. In this case, the user sets the desired angle on the LCD menu and continues printing. The system ends the current trace with the defined corner. Then the user places the tool on the other side of the angle and the system continues the trace (Figure 5.5, top row).

Second, when routing multiple signals, space often becomes an issue. To support creating compact designs, one might wish to add a line as close as possible next to an existing trace. *Print-A-Sketch* can automatically place a new trace in close proximity to an existing trace, creating parallel lines, which never touch. *Print-A-Sketch* does this by continually monitoring the distance to existing traces, and printing the new trace at a predefined distance. The user can specify the distance between traces directly on the device (Figure 5.5, middle row).

Lastly, not all printed traces should be connected to previously printed elements. Avoiding the creation of short circuits is crucial, as unintentionally created connections may result in irreparable damage. Here, *Print-A-Sketch* can support the user during sketching by observing the surface, detecting existing previously printed traces in the

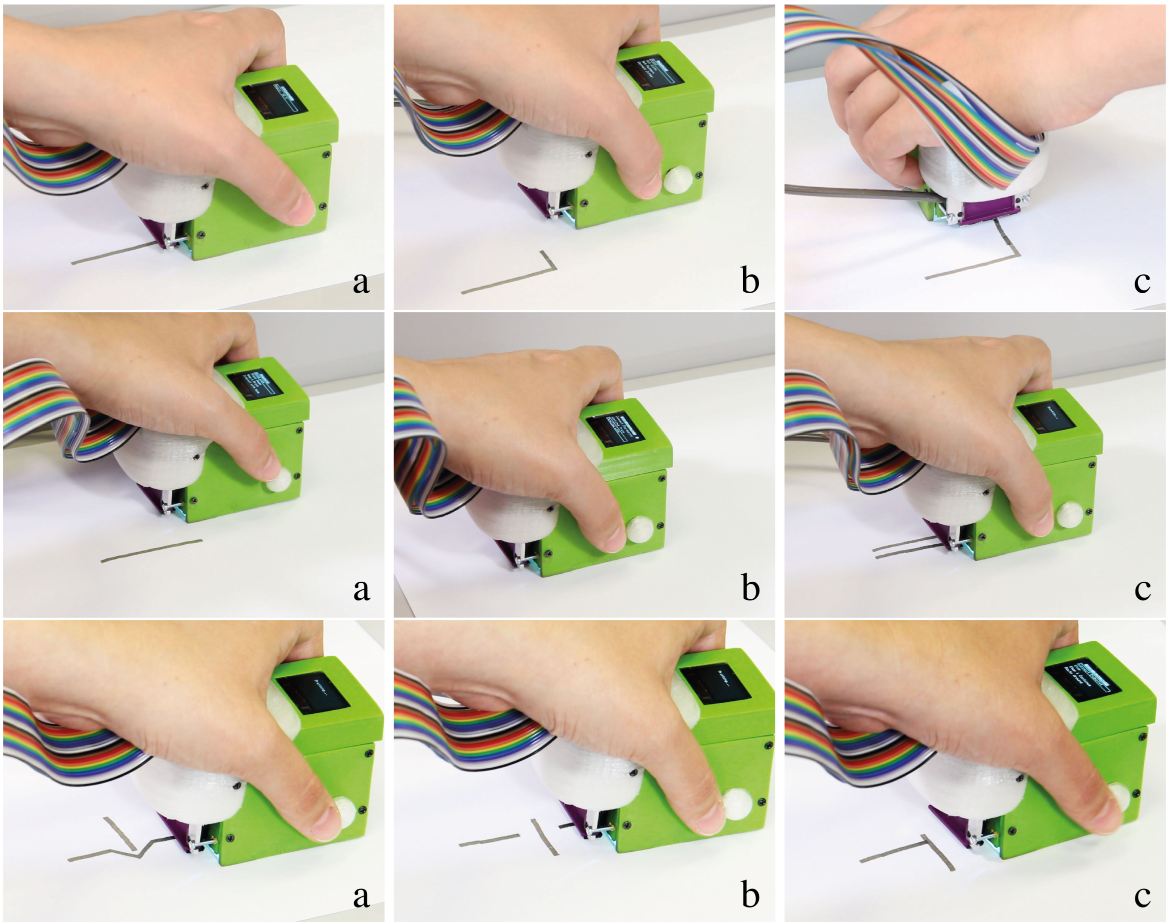


Figure 4.13 Top: Printing angle: a) defining the size of the angle on-the-fly, b) printing the angle, c) continuing the trace. **Middle**: Printing a trace parallel to a previously printed trace: a) selecting a previously printed trace, b) placing the printer on the existing trace and defining the distance between the traces, c) printing the new trace. **Bottom**: Various actions after detecting a previously printed trace: a) routing around the detected trace if there is enough space, b) stop printing before reaching the trace and continuing after the detected trace, c) forcing to connect to the detected trace.

print area, and alerting the user. By default, the system routes around detected objects if the size is less than the width of the print head (bottom row, Figure 5.5 a). If routing around is not possible, the system stops printing before reaching the existing object or trace (bottom row, Figure 5.5 b). Finally, the user can override the action and force the printer to connect to the detected trace (bottom row, Figure 5.5 c) using the buttons on the printer.

Accuracy of Adaptive Printing

We performed a technical experiment to investigate the accuracy of detecting a pattern on a surface and registering subsequent prints to it. We first printed five squares with dimensions ranging from 4 to 12 mm (2 mm intermediate steps) on office paper. Then, we moved the handheld tool around the surface and whenever a square was detected, the printer printed a crosshair over its center. Next, we measured the offset between the center of the crosshair and the center of the rectangle. The results indicate that the printed crosshairs are sufficiently close to the target; the mean offset was 0.478 mm (SD: 0.376) and the maximum offset was 0.813 mm.

4.4.3 Integration of Electrical Components

In most cases, traces alone do not make an interactive interface. A functional circuit needs to connect to sensors, actuators, and integrated circuits (ICs). To make this as effortless as possible, *Print-A-Sketch* supports two methods of interfacing with components:

Adding Footprints from Component Library

Print-A-Sketch contains a basic library, which contains footprints of common electronic components and printable components. In our current implementation, the component library is stored as bitmapped footprints. Users can navigate these and select them on the device. In addition to common footprints, printable components include capacitive sensors (capacitive button, slider, and pressure sensor) and resistive sensors (stretch and bend sensor, moisture sensor, and level sensor). Some examples are depicted in Figure 4.14, top. Note that resistive elements need to be calibrated to the target surface. The resistor shown in Figure 4.14 b works by increasing the trace length. To achieve higher resistance, the amplitude of the zig-zag pattern is increased.

Creating Footprints from Physical Parts

Additionally, for minimal disruption of manual sketching, many physical electronic components can be scanned using the RGB camera. Using basic edge detection, pins can be identified. Then appropriately sized and spaced pads are automatically generated. This feature is not package type dependent, however, it has only been tested on SIP, DIP, and SOIC packages (see Figure 4.14 bottom).

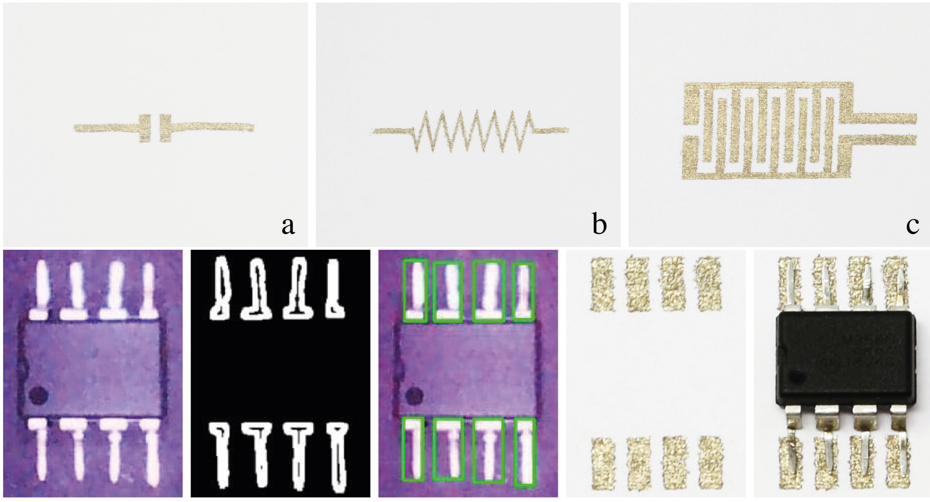


Figure 4.14 Top: (from left to right) Footprint for a 2835 component, printed resistor, moisture sensor. **Bottom**: IC footprints can be created on-the-fly by scanning the component to be used. (From left to right) RGB image, detected outlines, generated footprint, printed footprint, and printed footprint together with surface-mount IC.

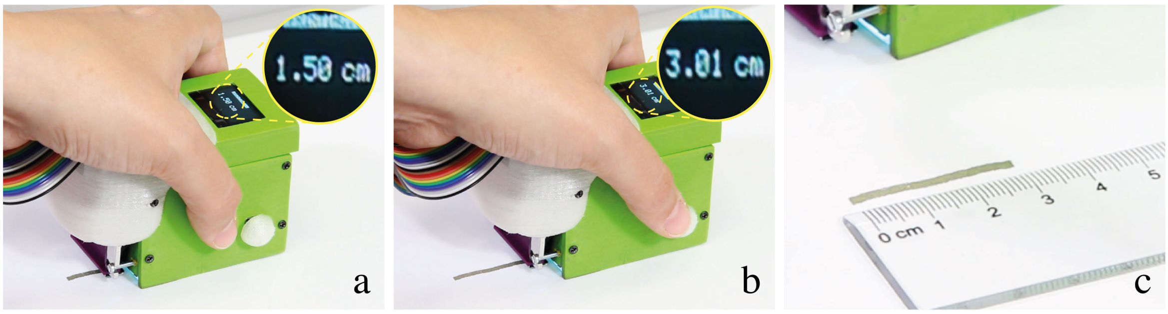


Figure 4.15 *Print-A-Sketch* as measurement tool: a-b) the display visualizes the length of the printed trace, c) shows the accuracy of the printed trace.

4.4.4 *Print-A-Sketch* as measurement tool

As *Print-A-Sketch* already continuously monitors the optical flow sensor for detecting speed and determining inking rate, it is also possible to use this information for measurement. The user can use this feature to measure the length of printing traces or spacing between components on the fly. The system guides the user by showing the measured value on the LCD (Figure 4.15).

4.5 Printing on Everyday Objects

Commodity desktop printers allow the user to manually define the substrate material (e.g., office paper or glossy paper). The printer's driver will then adjust the amount of ejected ink to best match the substrate's surface properties. A scenario involving a handheld printer may take up this principle but extends it to include more diverse materials, for instance, substrates that would not commonly be used in a desktop printer. Moreover, and contrary to a desktop printer, the user may sequentially print on multiple different materials, even during a simple trace. In this section, we discuss how continuous monitoring of the material and dynamic adjustment of print parameters allow users of *Print-A-Sketch* to freely sketch conductive patterns on everyday surfaces and objects. Moreover, we present results from an experiment that confirms that *Print-A-Sketch* prints conductive traces on a wide range of everyday materials.

4.5.1 Matching Print Parameters to Materials

The area that a jetted droplet covers on the substrate's surface vary depending on the substrate's ability to absorb ink. This has a direct effect on the conductivity of a printed pattern. As shown in Figure 4.16 top, droplets of a constant amount of ink may create end-to-end conductive traces on some type of substrate (top left), whereas on a more strongly absorbent substrate, ink drops are more strongly absorbed and therefore cover a smaller surface area, which creates a discontinuous pattern (top right). However, if the droplet size is too large, the ink will run, and detailed prints – such as the moisture sensor in Figure 4.14 c – will fail. Therefore, to ensure conductivity on all materials, the size of the ink droplet needs to be adjusted to match the material's properties. To this end, the electrical signal that activates the piezoelectric actuator of each nozzle is modulated [46]. Firing a single pulse results in dispensing one droplet (Figure 4.16 top). Larger droplets can be generated by firing a rapid burst of multiple consecutive pulses. This dispenses several droplets, which in turn results in formation of one single, but larger dot on the substrate (Figure. 4.16 bottom). Generally, the size of ink droplets needs to be larger when printing on absorbent substrates (e.g., plain office paper) compared to less absorbent substrates (e.g., glossy photo paper). Our implementation varies the size of droplets in a range between 1 and 10 droplets per dot or pixel[4].

The Computer Vision community has demonstrated robust and advanced techniques for optical material detection from image data [11; 37; 221]. Inspired by this work, we trained a KNN classifier with the sensor data from materials that we used for the evaluation and application scenarios. The handheld device displays the detected material and adjusts printing parameters based on a lookup table (Figure 4.17). Our current setup classifies

[4]In our implementation all printable images are stored as bitmaps with a width of 128 pixels, each pixel corresponding to an individual nozzle of the printhead. Therefore we maintain a 1-to-1 correspondence between dots and pixels

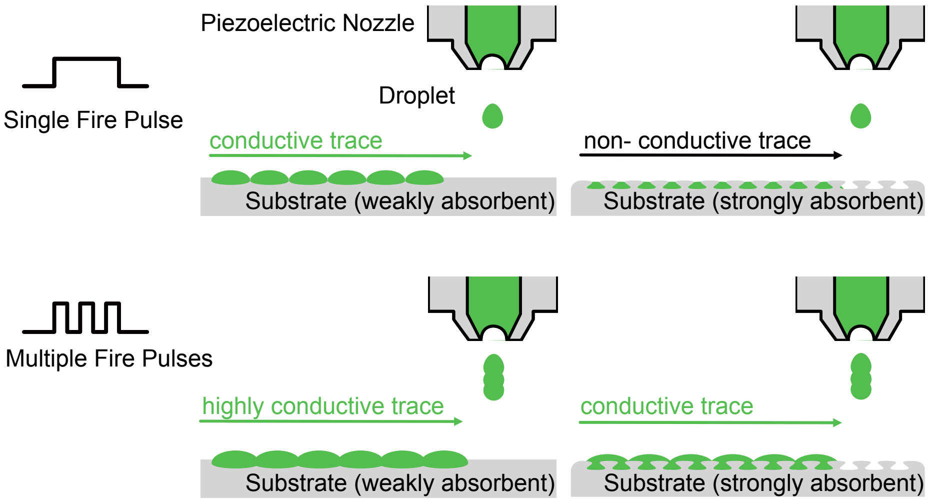


Figure 4.16 Increasing the number of pulses of the fire signal results in the creation of multi-drops and consequently larger drops. If drop sizes are too small, resulting traces will not be conductive. If drop sizes are too large, the ink will run, and detailed prints will fail.

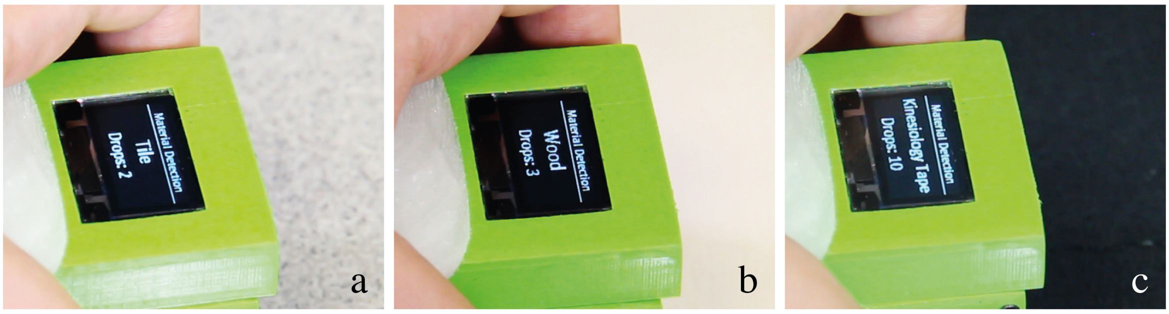


Figure 4.17 Adjusting the printing parameters to the detected materials: a) tile, b) plywood, 3) kinesiology tape.

only a subset of materials: office paper, glossy photo paper, tile, plywood, kinesiology tape, felt sheet, and natural cork. The current implementation serves as a proof-of-concept, it demonstrates the principle feasibility of dynamically adjusting print parameters based on automatically detecting the substrate material.

Knowing about the material and its mechanical properties do not only provide the basis for adjusting print parameters. Future implementations of *Print-A-Sketch* could automatically adjust the printed patterns themselves. For instance, when printing on more stretchable substrates that undergo more deformation, straight conductive traces could be automatically replaced with horseshoe designs. Or when printing on very demanding substrates, traces could be made wider to further increase their end-to-end conductivity.

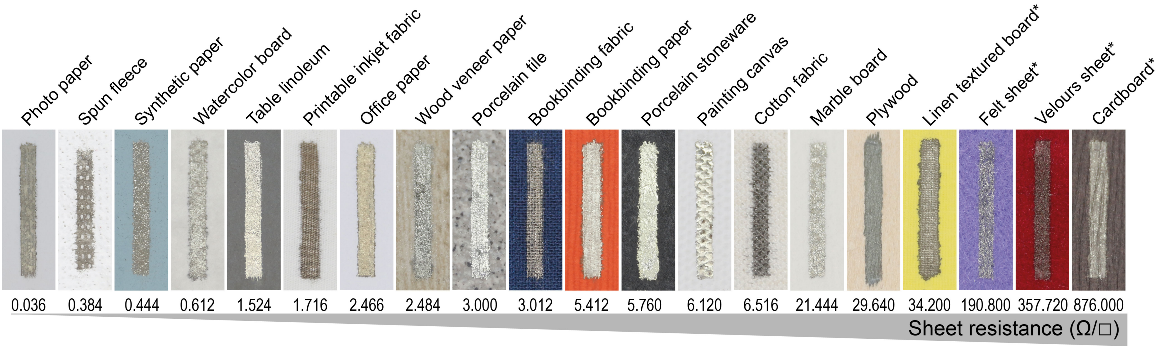


Figure 4.18 Sheet resistance of silver traces on different substrate materials, with single-drop printing.

4.5.2 Conductivity of Prints on Everyday Materials

To validate that *Print-A-Sketch* can print functional designs on many everyday objects, we investigated the conductivity of printed traces on various substrate materials. We investigated the effect of substrate materials, the number of droplets per pixel, and trace width.

Effect of substrate material

We selected a set of sample materials that are common on everyday objects and frequently used for prototyping (e.g., paper, cardboard, textile, plywood, and ceramics). We ensured that the selected samples cover a variety of material properties (i.e., surface texture and absorption). The tested materials are presented in Figure 4.18 and Figure 4.19.

On each material shown in Figure 4.18, we printed a trace of 20 mm length by activating 16 nozzles and firing a single pulse per byte of data (hence dispensing small ink droplets). We cured traces at 100 °C for 10 min. We used an electric oven (Sage, BOV820 BSS) to speed up the process by curing multiple samples simultaneously. As suggested by [98], we investigated the conductivity in terms of sheet resistance. To do this, we measured the resistance R of each sample using a Fluke 175 multi-meter, and then calculated the sheet resistance per square using the formula $R_s = R * (width/length)$, and averaged the results for each material.

The results are presented in Figure 4.18. As can be seen, most printed traces exhibit high conductivity. However, a handful of materials, such as cardboard, felt and velour sheets, result in very low conductivity. We show in the following how increasing the size of ink droplets can boost their conductivity.

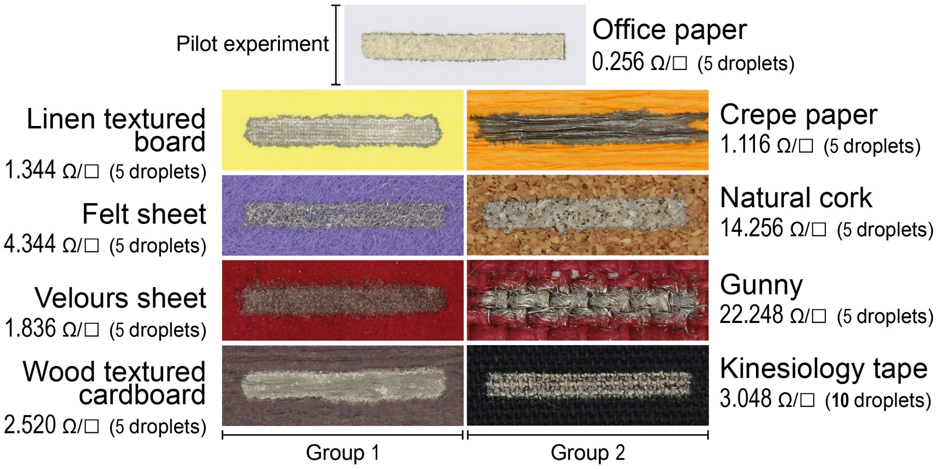


Figure 4.19 Improved sheet resistance with increased droplet size on highly absorbent substrate materials.

Effect of number of droplets per pixel

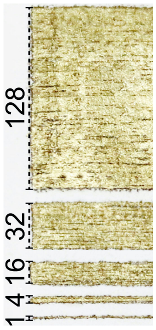
We next examined the effect of adjusting the number of droplets per pixel on conductivity. This promises to enhance conductivity on absorbent surfaces. For the first pilot experiment, we selected office paper as a frequently used material that is highly absorbent. Traces of 20 mm length were printed with 16 active nozzles, jetting different numbers of ink droplets (1, 2, 3, or 5 droplets for each pixel data) We printed 3 samples per condition, resulting in a total of 12 traces. The printed samples were cured at ≈100 °C for 10 min using a household iron (Bosch TDA2329). Our findings revealed that the conductivity considerably increases with increasing the number of droplets, ranging from 2.466 Ω/□ for 1 pulse to 0.256 Ω/□ for 5 droplets per pixel.

Based on this finding, we opted for the best-performing condition with 5 droplets for investigating more materials. We selected eight materials: four materials that have the highest sheet resistance in the previous experiment (group 1) and four materials that failed to become conductive in the previous experiment (group 2). On each material, we printed a 20 mm long trace with 16 active nozzles and 5 droplets per pixel, cured them as described above, and calculated the sheet resistance.

Figure 4.19 depicts the results. For all but one material, jetting 5 droplets substantially enhanced the conductivity compared to single-droplet printing. For the four materials that were part of the above evaluation, we observed an impressive 25 to 348-fold decrease in sheet resistance which belongs to linen texture board and wood textured cardboard, respectively.

For instance, the sheet resistance of wood textured cardboard, which had the highest sheet resistance in the previous experiment, has decreased from 876 Ω/□ to 2.52 Ω/□.

Table 4.1 Sheet resistance of traces with various widths on glossy and absorbent substrates (photo shows printed traces with five different widths on an absorbent sheet - numbers on left are nozzle counts).



Nozzles Max (128)	Glossy Sheet Width (mm)	Glossy Sheet Sheet Resis. (Ω/□)	Absorbent Sheet Width (mm)	Absorbent Sheet Sheet Resis. (Ω/□)
128	16.7	0.105	17.6	2.76
32	4.20	0.043	4.60	2.32
16	2.10	0.036	2.30	2.47
4	0.50	0.039	0.60	3.33
1	0.10	0.025	0.15	50.6

However, the trace printed on Kinesiology tape was not conductive with 5-droplets printing due to the highly porous composite material; we printed another sample with 10 droplets per pixel, which showed good conductivity.

Effect of trace width

To evaluate the effect of trace width and identify the minimum width of conductive traces, we printed 5 samples of 20 mm length and varying width (1, 4, 16, 32, 128 active nozzles, corresponding to 0.1–16.7mm width) on two different substrates using conductive silver ink [142], see Table 4.1. The substrates were selected from two different categories: a glossy paper which has a very smooth surface (Epson photo paper, 260 μm thick), and a highly absorbent sheet (standard office paper). For each pixel of data, the nozzles were fired once. The test was repeated 3 times and then the printed samples were cured at ≈100 °C for 10 min using a household iron (Bosch TDA2329).

Table 4.1 summarizes the results. We found that with the same number of active nozzles, printed traces are wider on office paper than on glossy paper. Furthermore, due to the high absorption of office paper, the conductivity is higher on glossy paper than on office paper. The sheet resistances on the glossy sheet range from 0.025 Ω/□ to 0.105 Ω/□ and are close to the reported value for silver ink (Metalon, JS-A102A) [142] which is <0.100 Ω/□. A minimum trace width of 0.1 mm is achievable on the glossy sheet by activating a single nozzle, resulting in a low sheet resistance of 0.025. In contrast, printing with a single nozzle on the absorbent sheet was challenging and resulted in the highest resistance (50.6 Ω/□). By activating 4 nozzles, resulting in a trace width of 0.6 mm, the sheet resistance decreased considerably to 3.33 Ω/□.

4.6 Validation

To demonstrate the practical feasibility of our technique and its applicability to the sketching of circuits and sensors on everyday surfaces, we present four functional application examples fabricated with *Print-A-Sketch*. These show various types of sensors and circuits realized on diverse materials that are commonly found in a household. Figure 4.20 shows the prototypes. In addition, we present findings from a hands-on case study with novice users.

4.6.1 Application case 1: Moisture sensor on a floor tile

The first application example demonstrates adding smart home functionality to a room's tiled floor. We designed and fabricated a customized moisture sensor on a porcelain tile (Figure 4.20a). When water comes in contact with the sensor, the resistance changes and the buzzer sounds an alarm. One use case of this application is detecting washing machine leakage. For the fabrication of the interactive tile, we selected a moisture sensor from the library of components and then printed the sensor, the footprint of the buzzer, and conductive terminals. Next, we created conductive lines that connect the buzzer to the terminals. The handheld tool helped by automatically printing a visually pleasing pattern, instead of straight lines, and by automatically connecting the line endings to the respective endpoints. Finally, we attached the buzzer using conductive paste and connected the terminals to an Arduino microcontroller using copper tape. Circuit elements that should not be in contact with water can be covered with isolation spray or silicone.

4.6.2 Application case 2: Interactive yoga mat

To demonstrate use on personal objects with demanding materials, we realized an interactive yoga mat that features four capacitive buttons for controlling the playback of instructional videos (Figure 4.20b). The mat is made of natural cork and is flexible. When the yogi desires to control the video, she can use her toe, without changing her pose, to pause, resume or navigate within the video. For printing the design on the mat, we loaded the design in the backend interface and used the *stamping* technique. The traces are printed using *routing* feature. A D1 mini microcontroller, attached to the backside of the mat, wirelessly communicates with the video player running on a tablet computer.

Figure 4.20 Application examples demonstrate the practical feasibility and applicability of our technique on everyday surfaces: a) moisture sensor on a floor tile, b) interactive yoga mat, c) wearable stretch sensor on kinesiology tape, and c) capacitive slider on a table lamp.

4.6.3 Application case 3: Wearable stretch sensor on kinesiology tape

To demonstrate rapid integration of sensors for textile wearables and inspired by sketching on-body interactions [209], we implemented a custom stretch sensor on a kinesiology tape (Figure 4.20c). The sensor can be attached to different joints of the user's body to detect flexion and extension. The terminals of the sensor are connected to an Arduino using standard copper cables. Stretch is detected by observing changes in resistance.

4.6.4 Application case 4: Capacitive slider on a table lamp

Next, we aimed to turn a conventional table lamp into a customized interactive piece of furnishing. We printed a capacitive button and two sliders onto a lamp's textile shade. These are used to turn the lamp on or off, and change its color and brightness (Figure 4.20d). To create the visual design, we drew the full moon and crescents on paper and then used our handheld printer to scan and then we print the designs on the lampshade. The electrodes were then connected to the terminals and tethered to an Arduino microcontroller to control the lamp.

4.6.5 Hands-On Case Study with Novice Users

To gain additional insights into patterns of use, we conducted a hands-on case study with four novice users (3 females, and 1 male). Two of them were familiar with sketching, painting, and designing circuitry, having backgrounds in computer science (P1, female, 24) and electrical engineering (P2, female, 23). P3 (male, 30) was experienced in fabrication and DIY electronics; P4 (female, 62) was a language teacher with no hardware prototyping experience.

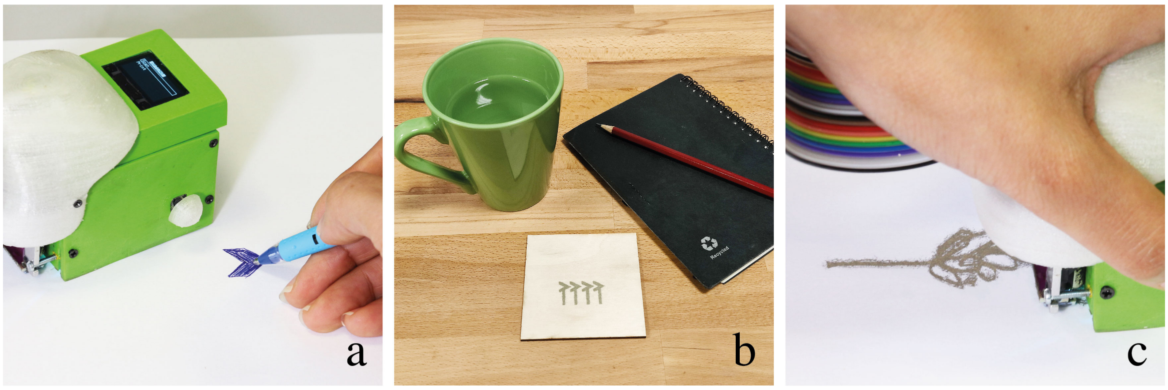


Figure 4.21 Images from user study: (a) drawing a customized electrode to scan & print, (b) Printed design based on the hand-drawn electrode. (c) free-form sketching.

After providing informed consent, we began the study with a short introduction to the project, handheld printer, and user interface, and gave participants time to practice using the device. Next, the participants were asked to perform a series of tasks from the simple to more complex: (1) print a shape; (2) scan & print a shape; (3) print a trace with custom width and pattern; (4) connect a trace to an existing shape. We then discussed the experiences and challenges they faced in a semi-structured interview. We continued the study with a brainstorming session and asked participants to think about a possible application case of this device in making their environment more interactive. They had the option to select a material from our sample box. The participants then proceeded to fabricate their idea with *Print-A-Sketch*, using the sketching interactions they preferred.

Results & Discussion.

All participants managed to complete the tasks and provided useful feedback. They commented that working with the device was simple and controlling the handheld device and moving it on the surface was smooth and easy to do. In addition, they commented on the usefulness of in-place printing using a handheld printer and its benefit for creating aesthetically pleasant prototypes. They stated that compared to traditional methods of designing and then printing, a handheld printer helps rapid prototyping and allows improvisation during fabrication. The possibility of printing and then embellishing the design was another interesting aspect for participants. Within an hour's session, all participants were able to make a functional prototype without our intervention. Participants, however, felt that we could provide better feedback, for example, P3 wished to have access to more information on the OLED display, while P4 – who had never before designed a circuit – felt that the breadth of options was overwhelming and would want to have more detailed guidance from the UI.

For their practical application case, they all decided to make capacitive buttons and sliders and used the *stamping* and *routing* features to fabricate the electrodes. P1 decided to create an electrode on her desk that uses capacitive sensing to detect touch contact, as a means to signal her when she is tapping constantly on the desk (Figure 4.21a). She fabricated the interface on a linoleum material, by printing a heart shape electrode from the library of components and then using the *connect-to* feature to print a serpentine pattern as a trace. P2 desired to make a slider on her bedside table to control the lighting of the room when she is in bed. Interestingly, she decided to customize the slider by sketching one custom-shaped electrode first and then used the *scan & print* feature to replicate the electrode to print a full slider on a sheet of plywood (Figure 4.21b).

Changing the TV program from the handle of a coach was interesting for P3. He used the *stamping* feature to print two capacitive buttons with a customized shape. Next, he used both *connect-to* and *motion correction* features to connect the traces to the printed electrodes. P3 was excited that the *motion correction* feature provides him with the ability to draw fine straight lines and highlighted that this feature is helpful for those who do not have sketching skills. Finally, P4 selected a diamond shape pattern electrode to fabricate an interface on her notebook. She intended to use this electrode to control the heating system at her home. While P4 was fabricating her electrode, we figured out that her understanding was that by pushing the handheld device, more ink would eject from the device, similar to pushing a fountain pen. We will consider this interesting finding for the next iteration of the device.

While most participants focused on moving the handheld device carefully and in a straight line, P1 decided to use the printer as a brush and freely sketched a floral electrode (Figure 4.21c). Finally, *Print-A-Sketch* not only supported novices in the fabrication of functional and visually pleasant prototypes but also facilitated thinking about making the surrounding environment more interactive.

4.7 Discussion

Our implementation is subject to several limitations. We discuss these in the following and identify opportunities for future work.

Materials and Printing. *Ink adhesion* shows the material compatibility. We observed the silver ink adheres well to diverse materials. We couldn't remove or wipe off the ink from highly absorbent materials and the ink is robust on glossy surfaces.

The JS-A102A nanosilver ink is water-based, which causes it to have higher *surface tension* compared to other types of inks, such as solvent-based ink. On substrates with low surface energy, such as smooth glass, this causes poor wetting. This issue could be addressed by applying an additive to the substrate before printing which reduces the surface tension and results in a better print quality [19].

In our current implementation, we use silver ink [142] which can be *thermally cured* at temperatures as low as 100 °C. A wide range of substrate materials tolerate this curing temperature; however, eliminating the heat curing step would enable printing on a wider range of materials and accelerate the fabrication process. We can report that printed traces on specially prepared and coated surfaces (e.g., Epson photo paper and Mitsubishi NB-RC-3GR120 [128]) sinter at room temperature. Promising avenues for future work are sinter-free inks (such as sinter-free gold ink [177]) or applying a sintering aid layer to the substrate to reduce the sintering temperature to room temperature [258].

Electrical circuits. For *electrical isolation* and to protect printed patterns from damage, we used isolation spray (WEICON isolation spray [236]). However, due to the crawling effect of the ink on the isolated layer, we were not able to print a second conductive layer. Therefore, our current approach is restricted to a *single-layer circuits*. In future work, we intend to address this by adding surface additives. This would allow the creation of layered designs, for example, to print multi-touch sensors [165].

While we have not performed any formal evaluation of *long-term conductivity*, we can anecdotally report that the conductivity of printed traces has not declined on tested materials (e.g., plywood, office paper, glossy sheet, and printable fabric) within a nine months period. As expected, we also observed that the printed traces are robust to *bending and stretching*. The response of inkjet printed traces to bending or stretching has been explored extensively in related research papers [26; 98; 99].

Iterative design and fabrication of a circuit may include *removing parts of a circuit* that was printed before [138]. We can anecdotally report that printed traces can be removed from coated paper (Mitsubishi NB-RC-3GR120 [128]) before heat curing using a sponge soaked in a mixture of water and polyvinyl alcohol (PVA). However, removing ink after sintering or from more absorbent materials requires additional investigation.

Open-loop printing. The size and form factor of the selected printhead allows *printing on curved surfaces and across edges*; however, our current implementation of position tracking using the optical motion sensor would not be reliable on non-planar surfaces. In future work, we plan to investigate alternative position-sensing techniques that allow for absolute positioning on a wider range of surface geometries. *Recognizing materials* from images is a topic of ongoing research [44; 193]. Our simple proof-of-concept implementation detects only a subset of all materials we successfully printed on. For the next iteration, we plan to extend the approach to a cover wider set of materials by sampling additional materials and tuning the printing parameters accordingly. The controlled lighting environment underneath the printer in combination with the dual data sources from the IR visual flow camera and the RGB camera suggests that this approach has promise.

To improve the visibility of printed patterns in the Figures and the video that accompanies this chapter, most examples were shown on white substrates. However, we demonstrated that the *blob detection feature* works on materials with different colors (e.g., black kinesiology tape, Figure 4.20c), visual patterns (e.g., non-solid colored floor tiles, Figure 4.20a and video figure), and material textures (e.g., uneven surface of natural cork, Figure 4.20b). We note that the accuracy of blob detection can be affected if the surface features visual patterns that are similar to the size or colors of the printed patterns. In future work, we plan to use the infrared spectrum [44] to improve the blob detection feature on surfaces with high-contrast colored patterns.

In our current setup, the RGB camera is located inside the handheld device, making the printed patterns in direct proximity to the printhead invisible to the user. To assist the user in precisely adjusting the printhead, the backend interface visualizes the live camera view (see Figure 4.5b), and the system automatically and precisely routes the printing trace toward the detected blob, if desired. In future iterations, we are considering displaying the location of the detected blobs on the OLED display. A simple alternative would be to make the casing of the handheld device out of a transparent material.

The *movement speed* of our handheld printer is restricted to 12 mm/sec. Of note, the printing speed of the Xaar 128 printhead is higher. The limiting factors are the Arduino system clock, the end-to-end latency between the microcontroller, optical sensors, printhead, and the backend interface. We plan to address this issue by replacing the Arduino with a faster microcontroller (e.g., Teensy 4.1, 600 MHz).

In our current implementation, the backend interface is running on a laptop and communicates with the Arduino via the serial port. In the next iteration, we plan to implement the backend interface in a Raspberry Pi and make the setup fully portable. Finally, creating a fully closed loop system that not only measures the environment of the trace but also samples the freshly printed trace itself to update inking parameters dynamically would be an exciting extension of this work.

4.8 Conclusion

In this chapter, we presented *Print-A-Sketch*, an open-source handheld printer prototype for sketching circuits and sensors. *Print-A-Sketch* combines desirable properties from manual sketching and functional electronic printing. Manual human control of large strokes allows for designing the overall interactive system while computer control provides fine detail, an interface to the miniature world of discrete components and integrated systems. Shared control of *Print-A-Sketch* enables sketching interactive interfaces on everyday objects which is especially exciting for objects so large or heavy that they could not easily be placed inside conventional printers.

We have shown that context-aware, dynamic printing can provide solutions for a number of challenges. Continuous sensing ensures quality prints by adjusting the inking rate to hand movement and material properties. *Print-A-Sketch*s sensing abilities also enable prints to intelligently adapt to previously printed traces for incremental and iterative sketching. Results show good conductivity on many materials (e.g.: 3 $\Omega/\square$ on tiles) and high spatial precision (traces are aligned with <0.5mm mean error), supporting on-the-fly creation of functional interfaces.

The following chapter explores how the confinement of handheld sketching tools can be extended beyond the reach of human arms.

5 RoboSketch: Mixed-Initiative Physical Sketching

Handheld tools, ranging from brushes and sculpting tools to cutter blades, offer the creative and practical maker an undisputed level of directness. Yet, purely manual work practices using such tools can be repetitive and cumbersome and are often constrained by the user's manual skills, precision, and physical abilities. With the emergence of ever more sophisticated means of digital fabrication, machines are taking over such tasks. While these machines have proven to be extremely useful to process users' intents without live intervention, delegating fabrication to the device in this way inhibits the inherently iterative nature of making.

Two streams of research have set out to tackle this problem from different directions: the first has proposed to add digital assistance to manual fabrication practices by augmenting handheld fabrication tools. Examples include hybrid carving [264], computer-assisted sketching [100; 163], 3D modeling [155], augmented airbrushing [198], and hybrid fabrication on the human body [56; 162]. While most of these approaches integrate directly with manual fabrication practices, their assistance suffers from a significant restriction: It is limited to the reach of the human arm, which prevents the device from carrying out fabrication tasks autonomously. It is always the user who has to lead the fabrication task.

The second research direction has investigated means to increase the interactivity of standard digital fabrication machines, for instance, by adding options for real-time design interventions to laser cutters [131] or 3D printers [154]. While these augmented machines can work more independently of the user and benefit from the precision and speed of high-end fabrication tools, they lack the ease and directness of in-situ physical practice with handheld tools.

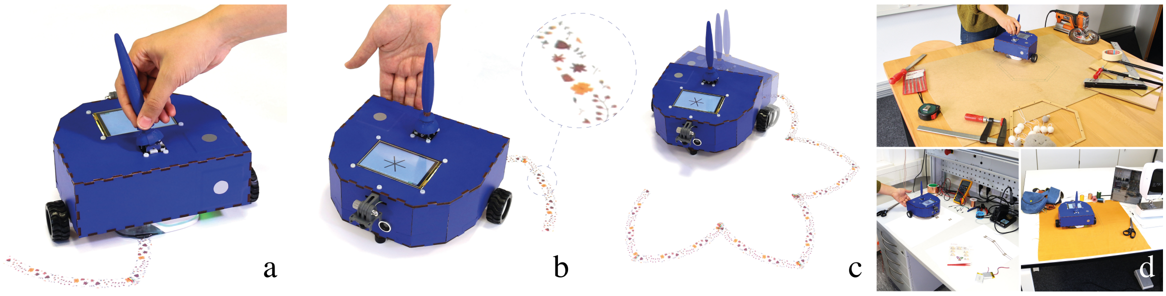


Figure 5.1 We created *RoboSketch* as an example of a new class of devices that combines the desirable properties of a computer-assisted handheld tool and an autonomous fabrication robot. The robotic high-resolution printer on wheels offers fluent transitions between manual, assisted, and autonomous modes. It assists the user in free-hand sketching (a). When triggered to switch to “autonomous mode” (b), it can roam freely to autonomously extend, complete, or refine the sketch (c). Application areas include fabricating electronic circuits, textile accessories, and woodworking (d).

We set out to integrate these worlds and propose a new class of devices that can be all three: a hand-operated manual tool, a computer-assisted handheld tool, *and* an autonomous fabrication robot. Such devices can assist the user where needed while in their direct proximity, but they can also be unleashed and roam freely, in order to solve some tasks independently. When done or called back, they return to the user and can again be operated in a manual or assisted mode.

In this chapter, we introduce *RoboSketch*[1]: a robotic printer on wheels with a joystick controller for manual sketching, capable of creating large-scale, high-resolution prints. It can be operated completely manually, inspired by a handheld brush *(manual mode),* but it can also provide interactive assistance during sketching *(assisted mode).* In addition, it can turn into an autonomous robotic device moving about for computer-generated sketches *(autonomous mode).* It is capable of operating on many surface materials, such as fabrics, paper, and wood, and with various inks including multi-color, UV, and conductive inks.

In the remainder of this chapter, we first introduce the approach of “handheld tools unleashed”: mixed-initiative physical sketching in which humans and machines work together proactively and fruitfully, unleashing the creative and unique benefits of handheld tools and robotic autonomy in concert. We discuss the emerging range of fabrication modes, from *manual* and *assisted* to *autonomous*, and highlight why seamless mode transitions are key in this context.

Next, we present interaction techniques to control such seamless mode transitions that are based on simple interactions well-compatible with sketching. These techniques

[1]This chapter is based on a publication presented at CHI’23 [164]. As the first author, I led the development of the robotic printer and sketching techniques, designed and fabricated the prototypes, and performed the evaluation.

support user-initiated and robot-initiated transitions between all modes, even while sketching a continuous trace. We also introduce a set of sketching techniques that benefit from these transitions to help the designer extend manual sketches, for instance, by repeating elements or upscaling a design, and to help revisit a sketch, for instance, to refine or color it.

We then contribute a proof-of-concept implementation of a functional robotic device, comprising a high-resolution print head that is capable of operating with a variety of surface materials and inks. It is based on a commercial handheld inkjet printer and a robotic platform equipped with various input controllers and sensors to be context-aware.

Finally, to validate that our approach is technically feasible and useful for physical sketching, and to illustrate that it can be applied in a wide variety of fabrication contexts, we present three application examples: (1) creating electronic circuitry, (2) creating sewing patterns on fabric, and (3) woodworking. In addition, we present our findings from a case study with seven sketchers. It uncovers flexible patterns of use and illustrates that mixed-initiative physical sketching can make computer-supported sketching more powerful and flexible.

In summary, the main contributions of this chapter are:

- the concept of a mixed handheld and autonomous device for mixed-initiative human–robot collaborative physical sketching that includes manual, assisted, and autonomous modes;
- interaction techniques to seamlessly move between modes and make use of the robot's autonomous capabilities to extend and revisit a sketch in the making;
- *RoboSketch*, a working prototype of the first computer-assisted robotic printer that supports mixed-initiative physical sketching across all its three modes, with capabilities to create error-preventing constraints, and validated to enable dynamic, context-aware sketching at high resolution and large-scale.

5.1 Handheld Tools Unleashed

We envision a new class of handheld devices to expand the scope of collaboration between humans and machines in creative design and fabrication processes, by combining the desirable properties of handheld tools with autonomous fabrication. Expanding upon Horwitz's notion of mixed-initiative interaction [80], we aim for tools that proactively contribute to the manual fabrication process whenever needed while allowing the user to continue working in a natural manner, but that can also contribute to the fabrication

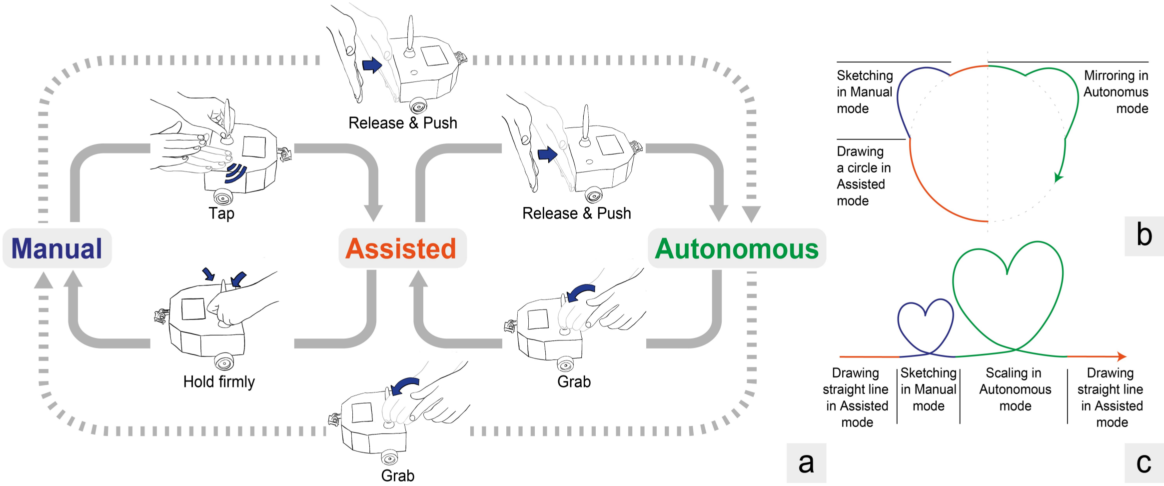


Figure 5.2 a) Transition of control sharing when the user initiates the transition. Examples of user-initiated transitions during a single stroke: b) Sketching a circle in assisted mode, overriding the constraints to add details manually, and then switching back to assisted mode to finish the first half of the sketch, before giving over the control to the robot to mirror the sketch across the vertical axis. c) Drawing a straight line in assisted mode, sketching a heart shape in manual mode, and then giving over the control to the robot to scale the design. Finally, taking over the control to finish the sketch with a straight line in assisted mode.

process completely autonomously if desired. With *RoboSketch*, we contribute a first and fully functional instantiation of this concept, demonstrating how a robot on wheels with a high-resolution color inkjet printhead can be used as a handheld tool for *manual* sketching, support *assisted* sketching, and can be "unleashed" to act as an intelligent robotic partner for *autonomous* drawing. *RoboSketch* addresses two key challenges:

Leveraging human and robotic skill sets Humans and robots partnering in a design and fabrication process would ideally leverage the unique skill set of each partner: While humans excel at generating creative ideas and can more easily adapt to a dynamic context and unforeseen events, robotic tools are capable of creating precise, high-resolution output and exact replicates at high speed. *RoboSketch* enables a variety of physical sketching techniques that demonstrate how human and robotic skill sets can complement each other. Using *RoboSketch* as a handheld tool, the user sketches out their creative vision before "unleashing" the device. *RoboSketch* is then able to autonomously expand upon the user's drafts by repeating patterns (e.g., leveraging symmetry), refining drafts (e.g., adding details), or by filling sketched-out regions with color. In addition, *RoboSketch* can offer the user to auto-complete their sketches (e.g., completing polygons), or offer creative completion by making use of AI to artistically elaborate on the user's input. Hereby, *RoboSketch* transcends the function range of existing computer-assisted fabrication tools such as FreeD [264] and Phasking [100]. While those tools allow the user to 'seize control' by overriding computer assistance, e.g., with a button-press, or by applying force, they still require the user's guiding hand to fabricate. They cannot fabricate autonomously beyond the confines of the user's reach. In contrast, our proposed approach enhances the scalability of the resulting designs and considerably enhances the extent to which the machine can act as a co-creator.

Flexibly shifting control back and forth Mixed-initiative physical sketching requires control shifts across the entire range from *manual*, where the user is in full control, over *assisted* with various levels of shared control, to *autonomous* mode, where the robotic tool is fully unleashed and sketches independently. To enable natural and efficient co-creation with both human and robotic tools iteratively contributing to the fabrication process, mode transitions need to be seamless. To this end, we developed a series of simple interaction techniques (Fig. 5.2) that enable fluent, *user-initiated* mode transitions throughout all modes at fabrication time: releasing the handle and giving the robot a gentle push signals the robot to continue on in autonomous mode (e.g., for elaborating on a user-created draft). In contrast, the user can solidify their grip ('Hold Firmly') to remain in control when in *manual* or *assisted* mode, or seize control by grabbing the robot's handle when in *autonomous* mode. *Robot-initiated* control shifts are necessary when the robot encounters contextual or environmental ambiguity and requires human assistance in *autonomous* mode. Here, the robot stops and blinks. Moreover, in *manual* or *assisted* mode, the robot proactively offers to take over control by making context-aware suggestions (e.g., auto-complete a shape) or by enforcing constraints (e.g., to prevent short circuits when sketching electronic traces with conductive ink). In summary, these techniques let the user access the full range, from handheld sketching tools to autonomous sketching robots with a single device, and even within a single stroke.

5.2 Sketching Techniques

RoboSketch offers a variety of sketching techniques and supporting tools to help designers, makers, and artists sketch out their initial idea, iteratively extend their idea, and revisit the composition to complete details. To enable natural and efficient co-creation by the human and the robot, these techniques fluently integrate manual sketching with computer-assisted handheld fabrication and autonomous fabrication.

5.2.1 Extending a Sketch

A human sketcher may require help when a design needs to be precise or symmetrical, contains repetitive elements, or when the canvas is large. Partnering with *RoboSketch* can help sketchers extend their creative vision while still maintaining a high level of precision and control.

Repeating pattern

Many sketches contain repeating patterns, which can be tedious and time-consuming to realize manually. The Repeat technique combines the expressiveness of manual drawing with support for repetitive sketching. Having selected the Repeat technique on the device's screen, the user starts by sketching the pattern (Figure 5.3,Ia) and then, in a seamless movement, pushes the robot toward the desired direction. This triggers the Autonomous mode; the robot takes over and continues printing the pattern autonomously and repetitively (see Figure 5.3,Ib), until the user takes back control by grasping the handle to continue sketching manually, or by holding the hand in front of the robot to stop the repetition at the desired position (Figure 5.3,Ic).

One of the main principles of design is achieving balance. This can be done by using symmetrical patterns. There are different manual techniques that can be used to create symmetrical drawings. For example, an artist may use tracing paper to trace a sketch and then flip it over to create the mirrored part. We provide assistance for creating repeated designs that are symmetrical around a central point or across an axis. As an example, to draw a precise *polygon*, the user first activates assistance to draw a straight line in manual mode (Figure 5.3,IIa). Inspired by [163], this makes the robot cancel out lateral hand jitter. After selecting the Polygon function on the device screen, the user draws the first polygon segment, then defines the number of sides of the polygon by tapping the robot the corresponding number of times (e.g., five taps to make a pentagon, see Figure 5.3,IIb), and then pushes the robot. The robot then sketches the desired shape autonomously (Figure 5.3,IIc).

For creating *multi-axial symmetries*, the user sketches the desired design in manual mode (Figure 5.3,IIIa) and then taps on the robot (or selects from the displayed menu, see Figure 5.3,IIIb) to set the number of radial axes across which the sketch is repeated. The user pushes the robot, and the robot finishes the sketch (Figure 5.3,IIIc).

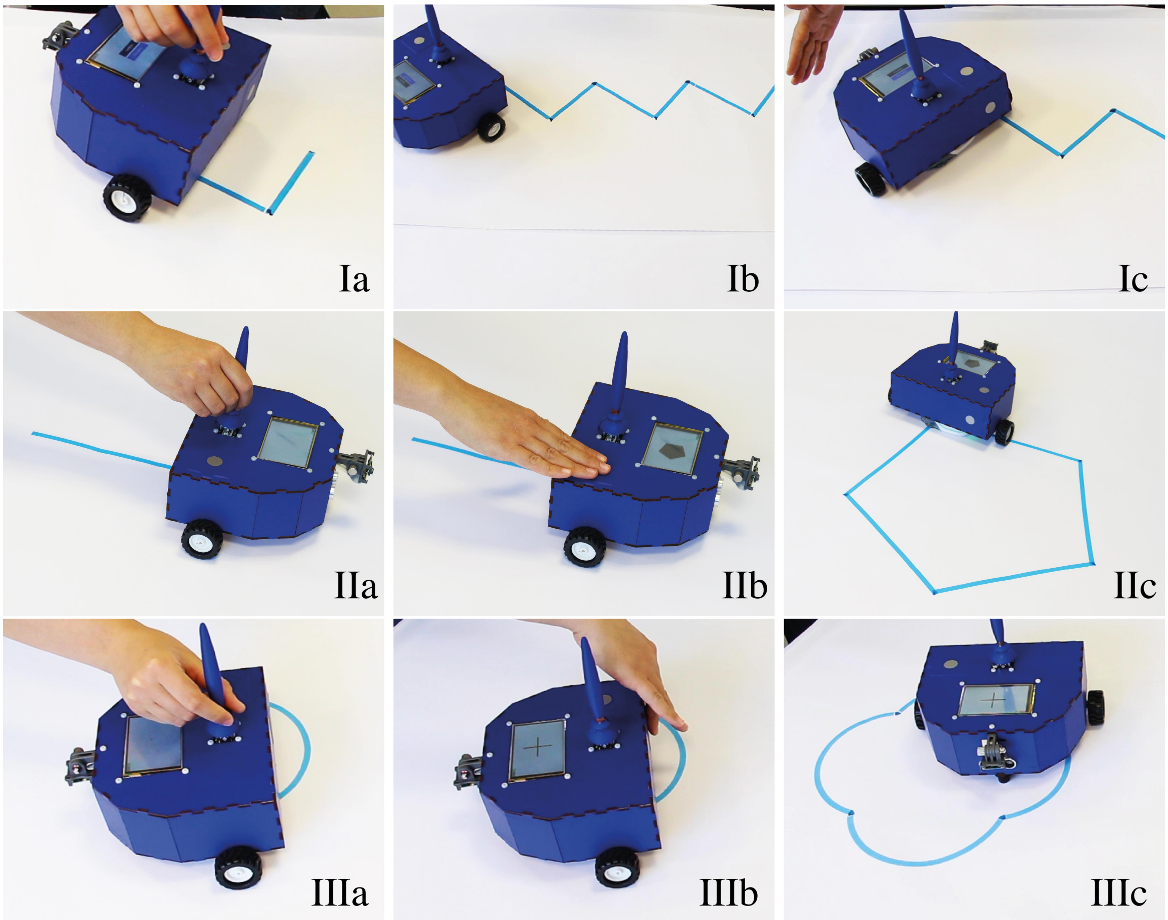


Figure 5.3 Repeating patterns: I) along a path, II) around a central point to create polygons, and III) across an axis to create symmetrical patterns.

Auto-completing shapes

To assist users to complete the current sketch quickly and precisely, *RoboSketch* provides an Auto-complete feature. When the user is sketching in manual or assisted mode, if the system detects the current shape, the prediction is shown on the display. If the prediction is correct and the user wishes to hand over control to the robot, the user simply releases the handle. The robot then autonomously completes the user's current sketch. Otherwise, the user continues sketching, and the predicted shape disappears or is updated with a new prediction. Our current implementation can recognize basic shapes (e.g., line, circle, square, and triangle) by inspecting the robot's movement trajectory. In the future, we will extend this feature to predict more complex shapes using a neural network [65; 119].

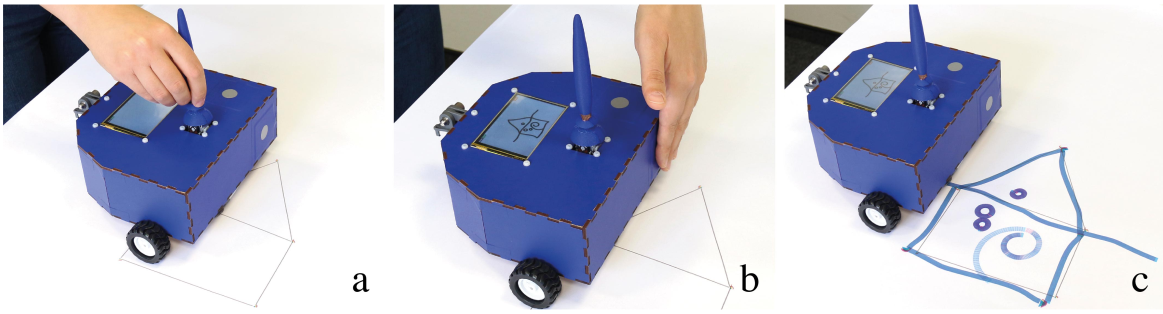


Figure 5.4 Creative completion: a) sketching the initial idea, b) accepting *RoboSketch*'s AI-generated suggestion for completion, shown on the display, c) robot sketches an artistic overlay over the existing sketch.

Creative completion.

Sketching is a medium for humans to visually express their thoughts, ideas, and emotions, often in an artistic way. On the other hand, recent advances in AI algorithms [183] have proven that they are capable of creating original visuals based on initial text and image input. By combining the advantages of both methods, humans and machines can co-create content and produce unique and personalized results. Pushing toward the machine end of co-creation, *RoboSketch* can realize new ideas based on the user's existing sketches (Figure 5.4b). We, therefore, use the recent implementation of the stable-diffusion model[2], based on [183], for image-to-image synthesis guided by a text prompt. In our current implementation, the user selects a Creative Completion function and starts sketching. We then query the stable-diffusion model with the user's current sketch as an initial image and with the text prompt *"line art miro style"* (100 steps of interference, prompt strength 85%) regularly. We post-process the resulting image with a standard auto trace algorithm (with center line option) [8] to create the paths for *RoboSketch* to print. Then, we show the result on the screen. When satisfied, the user pushes the robot to trigger autonomous mode, and the printer prints the AI-created image (Figure 5.4c).

Routing traces.

Sketching is an incremental and iterative practice. It is important to be able to pause and review a sketch, or return and add more detail. This implies that new traces oftentimes need to connect to existing traces and marks, and need to be precisely aligned. Some examples are closing a shape precisely, connecting elements in flowcharts and diagrams, or sketching conductive traces for electronic circuits. The Routing trace function assists the user in this task.

[2] https://github.com/CompVis/stable-diffusion

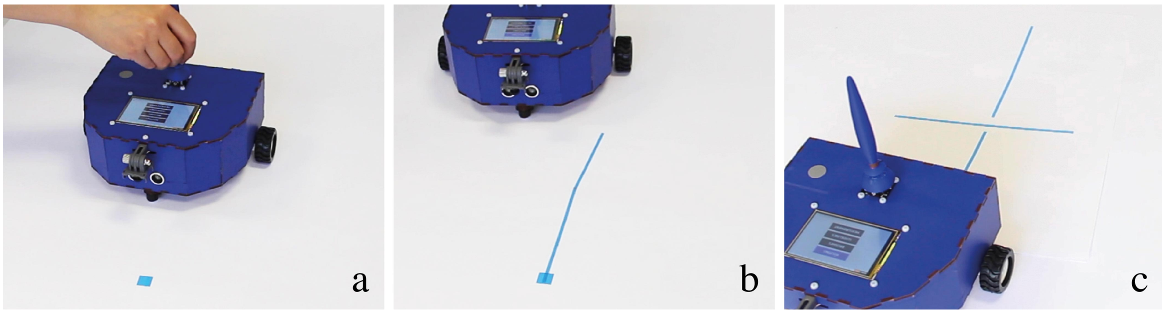


Figure 5.5 Routing toward existing mark: a) guiding the robot toward the desired direction, b) the robot detects the mark and connects the trace, c) the robot stops printing before reaching a trace and continues afterward, or can autonomously route around small visual elements.

When a user intends to connect a current trace to a previously printed mark, they manually sketch the trace toward the direction of the previous mark and then push the robot while letting go of the handle (Figure 5.5a). This triggers the autonomous mode. Now the system uses the built-in camera to monitor the surface and detect visual marks using blob-detection. After detecting the position of a printed mark, the robot fine-adjusts its direction so that the printer nozzles are aligned with the mark (Figure 5.5b) and keeps printing until it reaches the mark, precisely aligning the trace ending with the existing mark. If the user decides to take over control at any point (for example, to connect the current trace to another printed mark), they can grab the handle and continue sketching in manual mode. If the robot detects multiple marks, it connects to the closest mark by default, unless the user selects a different mark on the display. Optionally, to prevent undesired connection to previously printed traces in case of creating electronic circuits, the system alerts the user on the display when it is getting close to a printed trace. By default, the robot stops printing before reaching the trace and continues after crossing (Figure 5.5c). The user can select crossing or routing around the detected trace on the display.

Scaling.

It is common to scale a design to its intended size during sketching or do it even more flexibly using digital design tools. However, it can be difficult to scale a design when the final size is unclear or the canvas is large. *RoboSketch* enables creating sketches at a large scale, yet in place. For example, the user activates the Scale function, draws a small-scale design in manual mode (see Figure 5.6a), and then positions the robot at the desired location on the canvas. Then the user moves the robot from the lower left to the lower right of the desired bounding box to define the scale (Figure 5.6b). The user now releases

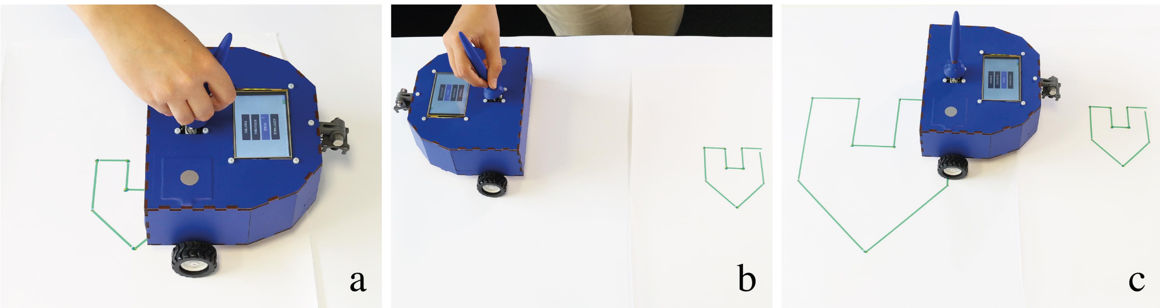


Figure 5.6 Scaling a design: a) sketching a small scale design in manual mode, b) defining the desired scale in assisted mode, c) robot sketches the scaled-up design in autonomous mode.

the handle and pushes the robot. The robot switches to autonomous mode and draws the design at the specified scale (Figure 5.6c). Scaling down works similarly.

Stamping.

Similar to prior work [163], the user can upload a vector graphic in the design tool, and then print the graphic by placing the device on the canvas and manually moving it in the desired direction. Extending beyond such manual stamping, we propose autonomous stamping in two variations: Firstly, the device can stamp a graphic along an existing contour, using line detection. Secondly, it can use stamping to extend an already existing marking with a graphic. To do so, the user places the device somewhere near the end of the existing marking. Using blob detection, the device identifies the marking's end, moves accordingly, and starts printing the graphic such that it connects to the existing marking. In all cases, the scale of the stamped graphic can be adjusted flexibly, provided it does not get wider than the printhead width.

5.2.2 Revisiting a Sketch

RoboSketch supports not only the creation of the overall structure but also the refinement and embellishment of a sketch.

Refining.

While it is fast and expressive to draw the overall structure and design of a sketch with a pen or brush, digital tools (notably, high-resolution printers) tend to be better at realizing detailed patterns and fine embellishments. Using this analogy, *RoboSketch* allows users to sketch the overall structure, before the device autonomously adds details to the design. For example, the user first sketches in manual mode (Figure 5.7a), then selects a

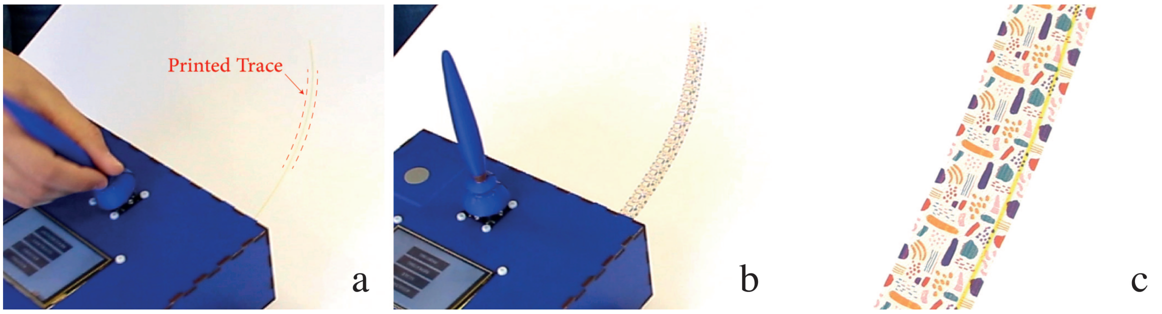


Figure 5.7 Refining: a) sketching a trace in light color, b) robot revisits and refines the trace with the desired pattern, c) close-up view of the pattern.

desired pattern from the list of patterns on the LCD menu, places the robot on the sketch (Figure 5.7b), and pushes it to trigger the Autonomous mode. The robot detects the trace using the built-in camera and prints the selected pattern along the trace (Figure 5.7c). At any time, the user can simply grab the handle to take over control and continue sketching. In our prototype, we considered that the robot follows a single trace to add details. In future work, we will consider more complex designs.

Beautification

Sketching is a natural way to create initial designs in the early stages of the design process. However, it is difficult to create precise shapes such as circles and right angles when sketching freehand. Beautification is the process of translating the hand-drawn and imprecise sketch to a regular and geometrically accurate design [229]. Inspired by sketch recognition research [211; 248], we used a $1 unistroke recognizer [247] to detect simple hand-drawn shapes and beautify them. The user first selects the Beautification feature from the display menu. They can then either use the inking or non-inking mode of *RoboSketch* to manually sketch the design; the system will beautify the design, and the robot will print a geometrically accurate result with a wider trace and darker color on top.

Coloring shapes.

After having created an initial line sketch, the sketcher may continue with painting to fill some shapes. *RoboSketch* supports coloring a shape with different tints and patterns. In doing so, the user selects the Painting mode from the display menu. Next, the user places the robot on a desired color or visual pattern; the robot records the pattern that is in its camera view (Figure 5.8a). Then, the user places the robot on the contour of a previously sketched shape, releases the handle, and pushes the robot to trigger autonomous mode (Figure 5.8b). The robot will then scan the shape's contour with the built-in camera, if

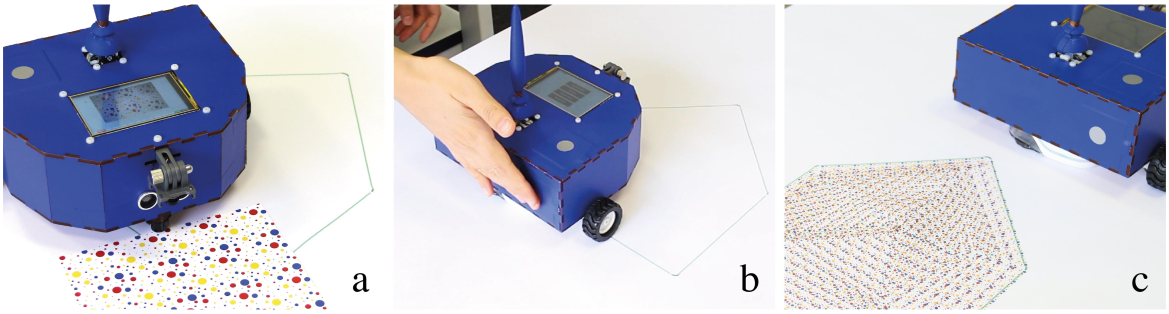


Figure 5.8 Coloring a shape: a) scanning the desired color/pattern, b) placing the robot on the shape's contour, c) robot fills the shape with the color/pattern.

required calculate a closed polygon, and paint the inner region by repeatedly printing the scanned pattern (Figure 5.8c). Our current implementation simply juxtaposes the scanned pattern; future implementation could use visual computing techniques to create a seamless pattern.

5.2.3 Supporting Tools

In addition to the sketching techniques for extending and revisiting a sketch, *RoboSketch* offers several supporting tools to enhance creativity, improve precision, and speed up fabrication:

Dynamic Custom Brushes

Artists use different techniques of brush movement to smoothly create different effects in a painting. They move the brush faster to create faded color, press the brush on the canvas to create a wider trace or choose a different color from the palette. Similarly, *RoboSketch* supports users to integrate these techniques into their sketching. For example, when the robot is in Autonomous mode, the user can take control for a brief moment by grabbing the handle to dynamically change the brush. Pressing the handle gradually will print wider marks (Figure 5.9a), moving the robot faster, by pushing the handle forward, will fade the colors (Figure 5.9b), and pointing the handle at the desired color while the color circle is displayed (see Figure 5.9c) will change the color. When satisfied, the user lets the robot continue sketching with the newly defined brush. Similar to other digital painting tools, *RoboSketch* also supports custom brushes (e.g. serpentines and zigzags).

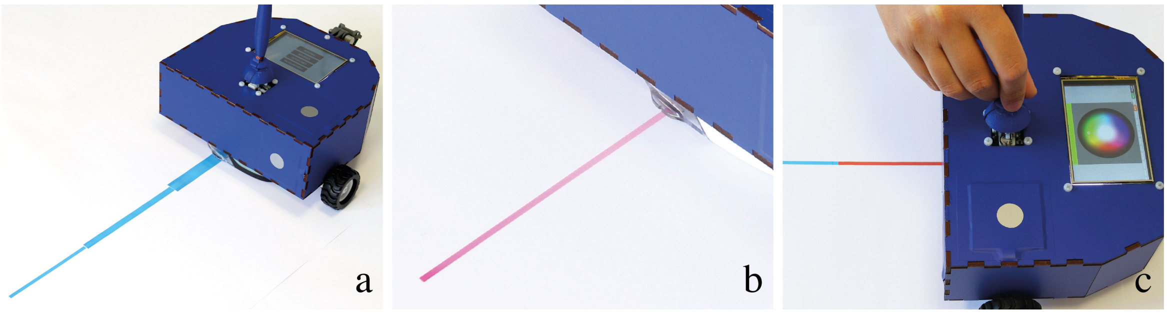


Figure 5.9 Custom brushes: a) pressing the handle creates wider traces, b) moving the device faster fades the color, c) and pointing the handle at the desired color changes the color on the fly.

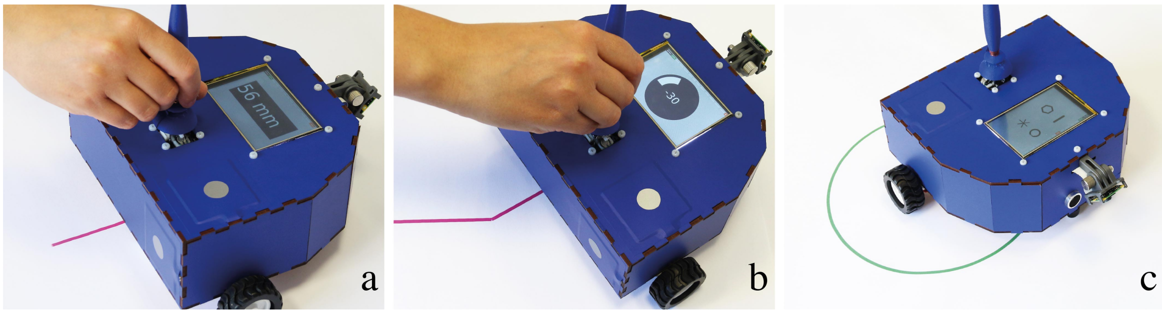


Figure 5.10 *RoboSketch* as a tool for a) linear measurement, b) angular measurement, c) and drawing guidelines.

Measurement Tool

To control the robot's motion, we use two encoders and continuously monitor their data. This data can also be used for measuring the length of the path traveled (*linear measurement*) (Figure 5.10a) or to print corners with precise angles (*angular measurement*) (Figure 5.10b). For example, to print marks on a certain distance (e.g., placeholders for screw holes), the user activates the linear Measurement tool, prints the first mark in manual mode, and then moves the robot while observing the distance traveled on the display, before printing the second mark at the desired position. To draw a corner with a precise angle, the user can grab the handle at any point, activate the angular Measurement tool, rotate the handle to define the desired angle, and then push the robot to continue drawing.

Guidelines.

Drawing guidelines offer valuable assistance for creating accurate and proportional sketches, provide guidance for outlining the design, and ensure that drawings are symmetrical and evenly balanced. *RoboSketch* supports designers in creating accurate guides by providing basic shapes (e.g., line, circle, polygon) and radial symmetry. As an example, the

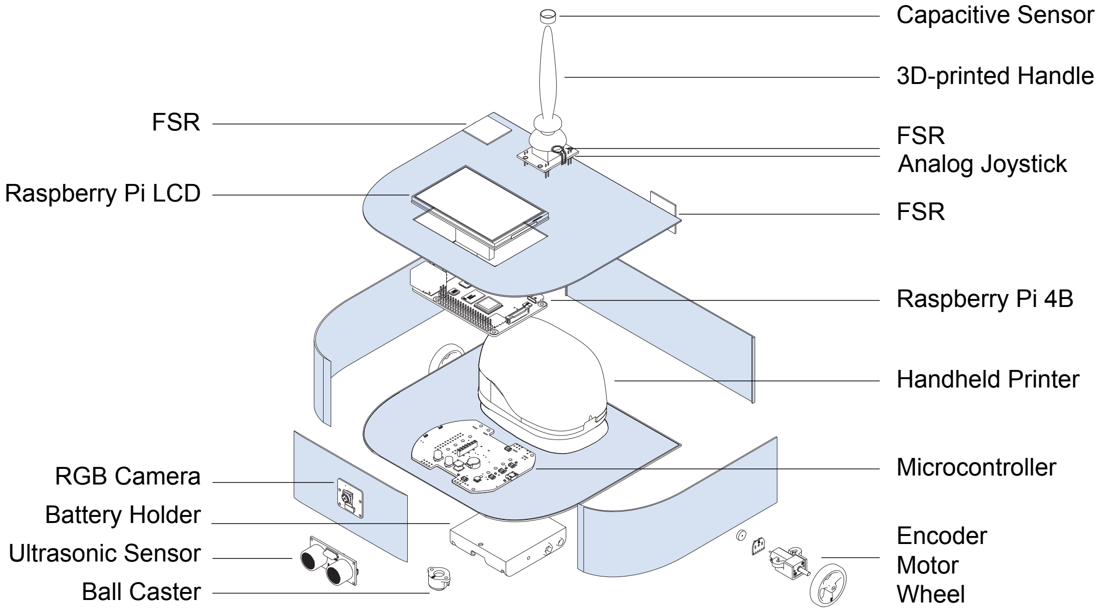


Figure 5.11 *RoboSketch* includes a mobile robot, a handheld printer, and sensors for sensing and tracking. A handle on top enables physical sketching with the device.

user can create a guide in the form of a circle by selecting the circle from the display menu, specifying the center point and radius with a stroke in manual mode, and then releasing the handle. The robot will then complete the task in Autonomous mode (Figure 5.10a). By using UV ink to print guides and then UV light to continue sketching, we can make the guides invisible in natural light. Alternatively, guidelines can be printed with a very fine width and light color. In the future, advances in ink technology may make it possible to erase the printed traces or print sketches that fade after a while.

5.3 Implementation

We now present the proof-of-concept implementation of *RoboSketch*. We first discuss the hardware system, then the user interface for controlling *RoboSketch*, and finally the implementation of interactions.

5.3.1 Hardware System

Robotic Base. The main components of *RoboSketch* are shown in Figure 5.11. Two micro metal gear motors (HP 6 V, 250:1) [159], controlled by two DRV8838 motor drivers, are used to move the robot in differential drive mode. The motors are equipped with magnetic encoders (12 CPR) [160], used to measure the distance traveled by the robot. They are tethered to an ATmega32U4 AVR microcontroller. The device moves at a maximum speed of 31 cm/s. The body consists of a laser-cut MDF case and measures 164 x 191 x 60 mm. 4 AAA batteries power the robot and provide about 8 hours of operation without recharging.

Sensors. An ultrasonic distance sensor (HC-SR04), tethered to the microcontroller, is used for detecting obstacles. A wide-angle RGB camera (OV5640) mounted on a stand is connected to a Raspberry Pi 4B. With an embedded Linux operating system and the use of OpenCV's blob detection feature, it monitors the robot's surroundings, detects previous marks, and provides a real-time video feed for debugging.

Printer & inks. *RoboSketch* contains a color handheld printer for high-resolution prints. We used COLOP e-mark [29], a commercial handheld thermal inkjet printer, which has a very compact form factor (111 x 76 x 72 mm). It is lightweight (225 g) and able to print on diverse absorbent surfaces (e.g., paper, cardboard, cork, textiles, and wood). With its 14.5 mm wide printhead, it allows for high-resolution prints (600 dpi) at a maximum printing speed of about 30 cm/s. The selected handheld printer allows changing and refilling the printer cartridge with various inks. Commercially available replacement cartridges comprise tricolor, black pigment, and UV ink. In addition, we have successfully printed conductive silver ink, in line with prior work that used inkjet heads for printing conductors [98; 163].

5.3.2 Software Implementation

To enable a rapid and convenient workflow, users are provided with a two-part user interface. The touch screen user interface (Figure 5.12a), embedded on the robot and implemented in Processing, facilitates direct and immediate interaction with *RoboSketch*. The user can trigger most functionality directly on the robot (e.g., selecting primitives, changing the brushes' pattern and color). Moreover, the display provides real-time assistance and shows the position of the robot relative to the traversed path. We used a 3.5-inch Raspberry Pi LCD [233], inserted directly into the Raspberry Pi board (Figure 5.11).

In addition, we implemented a backend interface in Processing that is running on a standard laptop (Intel Core i7-6700HQ CPU 4 cores at 2.60 GHz) with Windows 10

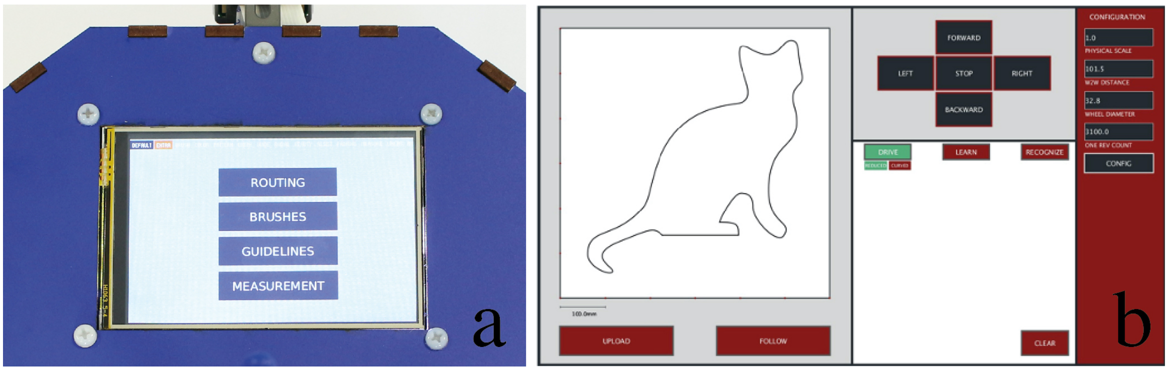


Figure 5.12 *RoboSketch* user interface: a) the user interacts with the device using the embedded LCD, b) the backend interface links all components and provides additional functionality (e.g., uploading a new design).

(Figure 5.12b). The backend interface allows debugging of the system and establishes a link between all the components: it communicates with the ATmega microcontroller via a Bluetooth connection to receive sensor data and control the motors and uses Wifi to communicate with the inkjet printer and Raspberry Pi.

5.3.3 Implementation of Interactions

RoboSketch enables physical sketching and direct manipulation of the robot using a handle. For this purpose, a dual-axis analog joystick module including a push button [3] was used and connected to the base microcontroller (Figure 5.11). To facilitate interaction, we 3D printed a brush-like handle out of PLA and replaced the original joystick knob with our 3D printed handle. We use relative mapping: pushing the handle more will make the robot move faster. By placing a small force-sensitive resistor (FSR) [201] between the tip of the handle and the push button, the device provides different levels of pressure on the handle, giving the user more flexibility when interacting with the robot. A capacitive sensor on the tip of the handle, made of copper tape, detects the presence of the hand. For detecting tap and push gestures, two square FSR sensors [202] were placed on the top and back of the robot (Figure 5.11) and then connected to the base microcontroller.

To support controlling the robot from a distance, the user can use a stylus and digitizer tablet [227] or a gaming controller [45] that communicates wirelessly (2.4 GHz) with the backend interface to control the robot remotely.

5.4 Validation

To demonstrate the practical feasibility and versatility of our technique, we present three application examples fabricated with *RoboSketch*. These show the use of sketching techniques and the transition between different interaction modes, in various domains of fabrication. We also present the results of a hands-on case study with artists and engineers.

5.4.1 Dandelion Art with Interactive Circuitry

Inspired by Jie Qi's Dandelion Painting [263] and to demonstrate how *RoboSketch* can support creative activities and facilitates the fabrication of electronic circuits, we created an interactive wall art that glows from behind (Figure 5.13g). The painting is made on three pieces of A3-size cold-pressed sheets and consists of two layers: the front is an artistic layer showing dandelion flowers, while the back contains the electronic circuit and LEDs [171; 172]. We started by manually sketching two lines to create the stalks of two large flowers. For sketching many small dandelion seeds, we uploaded a graphic of the seed and used the Stamping and Scaling features to freely print it at different sizes and orientations. For sketching the stalks of larger seeds, we uploaded a graphic of just the stalk and stamped them freely at different orientations (Figure 5.13a). To complete the seeds, we uploaded a graphic containing the seed's feathery bristles and switched to autonomous mode (Figure 5.13b), to let the robot identify the stalk's endpoints and autonomously print the bristles in the right places (Figure 5.13c).

To create the electronic layer, we first used Stamping to print the footprint of the LEDs on photo paper [48] with conductive ink (Figure 5.13d). We moved to autonomous mod (Figure 5.13e), using the Routing Trace feature, to let the robot connect the footprints with conductive traces (Figure 5.13f). Finally, the LEDs are placed on the footprints and both layers are attached to a wooden frame. All the traces are connected to LiPo batteries attached to the back of the canvas.

5.4.2 Creating Sewing Patterns on Fabric

Transferring a sewing pattern onto fabric can be a tedious task, often done manually with a pen, tracing paper, and previously cut templates because most textiles do not fit into a commodity printer. *RoboSketch* assists textile makers in creating customized cutting and sewing patterns on the fabric. As an example, we created a clutch bag from a piece of velvet fabric for the outside and linen fabric for the inside (Figure 5.14f). We first uploaded a graphic with the cutting and sewing pattern (Figure 5.14a). Next, we defined the appropriate scale and position of the pattern directly on the piece of fabric, using the Scale and Measurement features (Figure 5.14b). The robot then printed the pattern on the back of the fabric (Figure 5.14c). We repeated the previous tasks to create all the

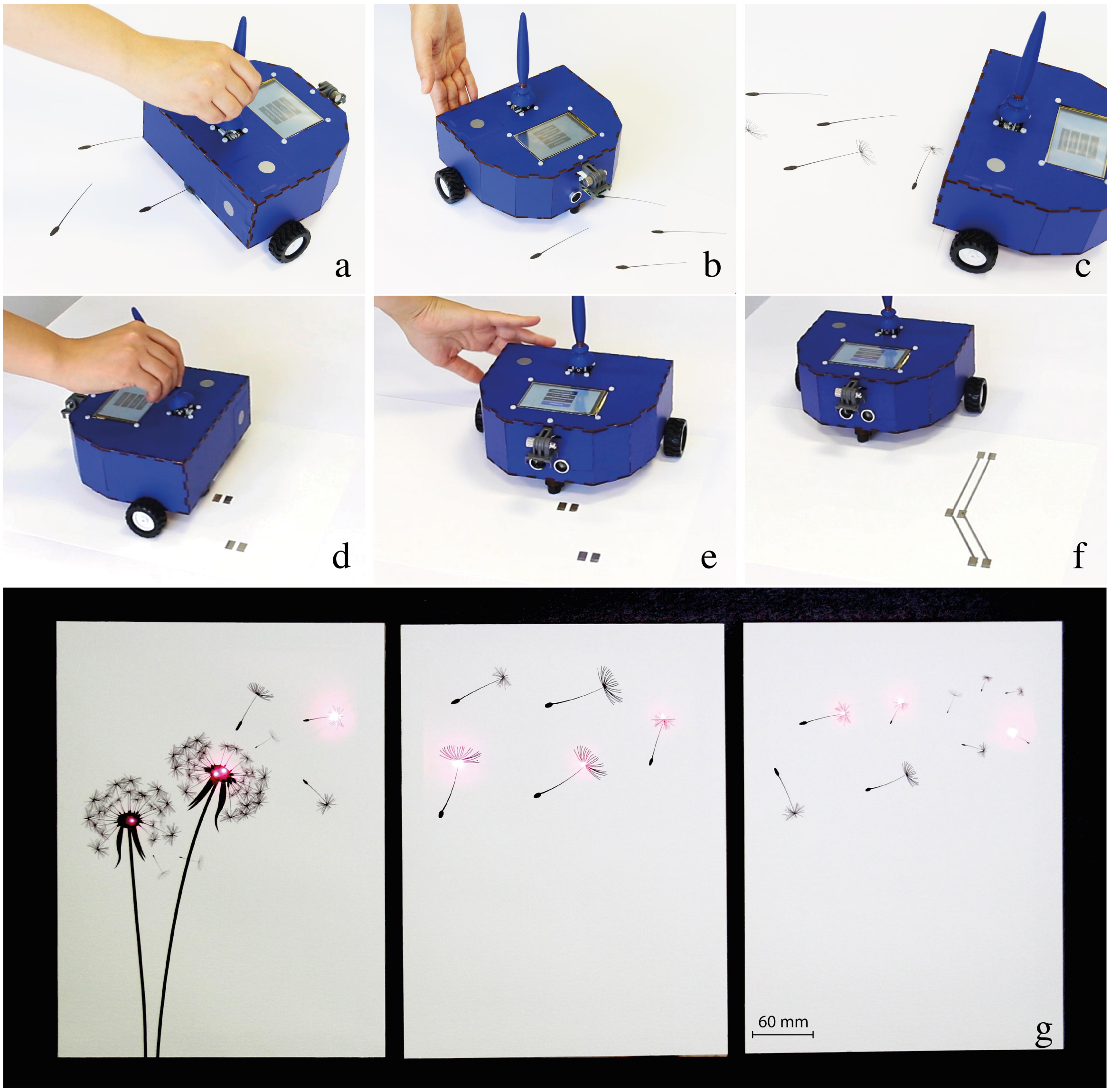


Figure 5.13 Interactive dandelion wall art: a) printing the seed's stalks using Stamping, b) pushing the robot to Autonomous mode, c) the robot autonomously prints the seed's bristles at the appropriate location, d) printing the footprints of LEDs, e) pushing the robot to Autonomous mode, f) the robots detects and connects the footprints, g) fabricated dandelion artwork glowing from behind.

pieces. Finally, we cut out the fabric along the traced line (Figure 5.14d) and sewed the piece together using a sewing machine (Figure 5.14e).

5.4.3 Assistance in Wood Working

Creating a precise and intricate design on a piece of wood is challenging. Craftsmen sketch the design on the wood with a pencil and use various measuring tools (such as

Figure 5.14 Creating sewing patterns on fabric: a) uploading the design, b) scaling the design on the fabric, c) the robot prints the pattern on the fabric, d) cutouts with sewing marks, e and f) fabricated clutch bag.

a ruler, protractor, and combination square) to create straight lines and precise shapes. *RoboSketch* facilitates crafting by assisting to sketch precise shapes and align screw holes on a piece of wood. As an example, we realized a wooden hanger for a crib on 3 mm thick plywood and then attached toys with strings (Figure 5.15f). To create the hanger, we selected the Repeating Pattern feature (polygon) and sketched the polygon's first segment on the wooden sheet in Assisted mode (Figure 5.15a). After defining the number of sides to six, the robot completed the polygon in Autonomous mode (Figure 5.15b). Next, we added marks for drilling holes where toys will be attached. We created the first and second marks at a 5 cm distance using the Stamping and Measurement tools (Figure 5.15c) and then switch to the autonomous Stamping mode, to let the robot repeat stamping the marks following the polygon (Figure 5.15d). Finally, we cut the plywood (Figure 5.15e) and attached the toys with strings.

5.4.4 Case Study

To gain a better understanding of *RoboSketch* in use, we conducted a hands-on exploration session with experienced artists, sketchers, and novices.

Participants

We recruited 7 participants: 3 artists from the College of Fine Arts, all female, aged 30 (A1 and A2) and 33 (A3), and experienced in a wide range of arts, including sketching, drawing, and painting with physical tools. The other 4 participants were engineers with backgrounds in embedded systems (P1, female, 22), e-textile (P2, female, 28), robotics (P3, male, 30), and soft robotics (P4, male, 31). Two participants were left-handed.

Procedure

We began the study with an introduction to the project, basic functionalities, and interaction with the robot, and gave participants time to practice sketching with our tool. They also tried different supporting tools, such as custom brushes and measurement tools (Figure 5.16a). Then, we continued the study by explaining the Manual, Assisted, and Autonomous modes and introducing the gestures for transitioning between these modes. Participants were then asked to perform a series of tasks to familiarize themselves with the transition of shared control: 1) repeating pattern (linear, polygon, and symmetry), 2) scaling, and 3) sketching in Assisted mode (straight line and within boundaries). We also gave them time to explore other features that interested them. We then discussed their experiences and the challenges they faced in a semi-structured interview. We continued

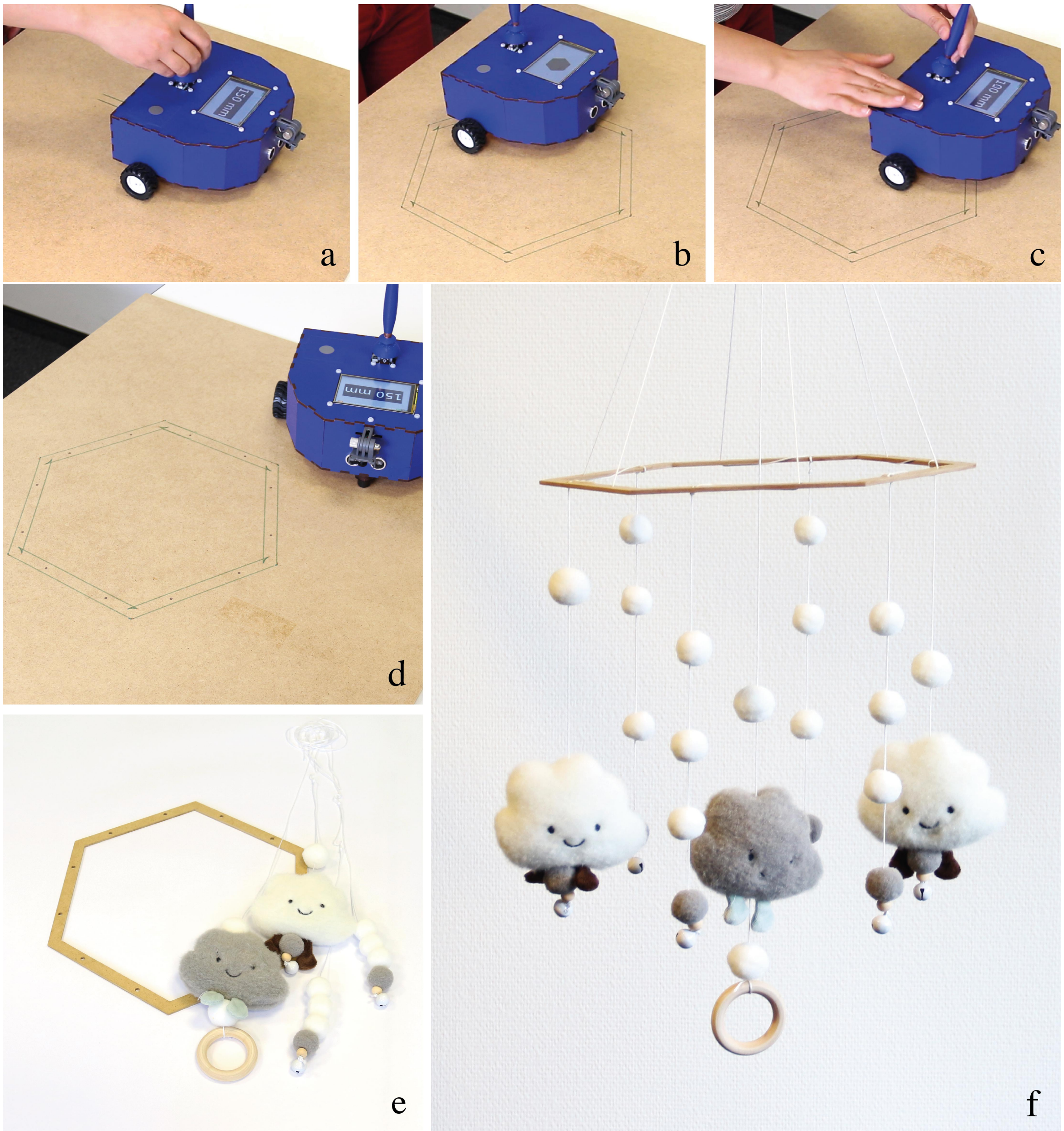


Figure 5.15 Assistance in woodworking: a) defining the first segment of the polygon, b) the robot completes the polygon, c) defining the mark for drilling holes, d) the robot completes printed the drilling marks, e) hanger cutout, and f) fabricated hanger for a crib.

the study by asking participants to create a drawing (one result is shown in Figure 5.16b) using their preferred sketching techniques (two participants did not finish this task due to lack of time). Finally, all participants were asked to complete a questionnaire about their experience and possible use cases of the tool. The sessions lasted about two hours, were audio-recorded, and photos and videos of key situations were taken.

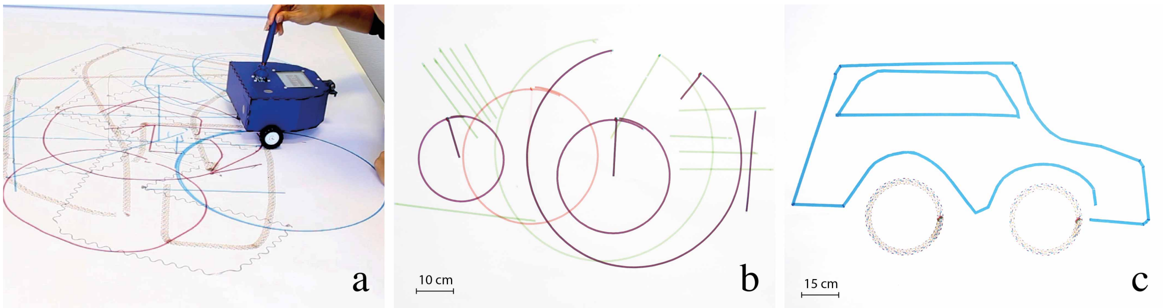


Figure 5.16 a) Participant during hands-on exploration session, b and c) drawing created by two of the participants.

Results & Discussion

All participants were able to interact with the device and provided valuable feedback. Most importantly, after less than one hour of exploration, they were able to sketch with the device without our intervention. In the following, we summarize the central findings.

Likert scale. As part of the questionnaire, we asked participants to rate the following questions on a five-point Likert scale: *How easy was it to use the device, and how likely would they use manual, assisted, and autonomous modes, and sketching techniques?* Overall, responses were positive to very positive (30 out of 35 responses were "likely" and "very likely"). Participants valued sketching techniques and various modes of interaction with the device, with autonomous (5 out of 7 very likely, 1 likely, and 1 neutral responses) and manual modes (3 out of 7 very likely, 3 likely, and 1 neutral response) being favored most.

Manual mode. Participants liked the ability to manually move the device, draw very consistent lines, and change the color, width, and patterns of traces quickly and on a large scale. P2 liked the idea of controlling a handle like a brush; P3 mentioned that *"the joystick design is comfortable to interact with"*, and A2 enthusiastically said, *"you only need one tool instead of many pencils"*. Interestingly, A1 wished for a longer handle to control the robot on the floor, to print sketches during an on-stage performance, and then requested to control the robot remotely with the remote control joystick. Similarly, A3 expressed her interest in sketching street art from far away with a remote controller. While all participants liked the concept of Manual mode, they also pointed out that the current size of the device is rather large for a handheld device. From our observation, after a few minutes of practice, the artists were able to move the device confidently and make freehand sketches; in contrast, the engineers were careful about parts of sketches that were hidden underneath the device and indicated that they needed more time to practice.

Assisted mode. All found Assisted mode very helpful, especially for drawing straight lines, geometric shapes, and keeping boundaries: *"Seemed like magic, merging of real-world and virtual borders"* [P4]. P2 stated that the assistance in keeping boundaries allowed her to focus on sketching without worrying about crossing boundaries. P3 decided to sketch a car when we asked him to create a drawing with the device. He mentioned that he is very untalented and uneasy in drawing by hand; however, using Assisted mode for drawing straight lines and basic shapes, he managed to draw a large-scale car on a piece of paper (150 x 110 cm). At the end of the session, he was satisfied that he could draw by hand for the first time (Figure 5.16c). P4 found this mode useful for drawing graphics and 2D CAD drawings that are difficult to sketch by hand. All participants except one (A1) preferred to be notified before receiving assistance from the robot.

Autonomous mode. All participants were enthusiastic about the robot moving autonomously and expanding their hand-drawn sketch: *"I liked Autonomous mode (...) I can just observe my drawing expand"* [A1]. They also mentioned that the Autonomous mode will allow them to repeat difficult shapes and patterns that were difficult or tedious to do by hand. For instance, based on her experience in drawing comics, A1 found this mode very helpful in scaling and repeating visual elements in comics faster and more accurately. P3 mentioned that for his project on metamaterials, he has to replicate similar patterns (e.g., cells) in different angles and scales to be able to analyze them. This device allows him to make faster and more accurate sketches for ideation and further discussion. He then continued sketching one of the cells and repeated it in a different direction. P1, who has been drawing mandalas for several years, expressed that autonomous mode can help her create a more customized and precise design. She then sketched half of a butterfly and used the Repeat function to mirror it.

Transition of control sharing. Participants valued the tangible interaction with the device and preferred to touch the robot to initiate the task rather than pressing a button on the UI. A1 said, *"I like the tangible interaction with the robot, it was a fluid movement between me and the robot"* and she continued *"I feel connected to the robot when I touch it"*. P3 indicated that the gesture metaphors are memorable. Participants learned the gestures quickly. We frequently observed that they began sketching in manual or assisted mode, then pushing the robot to extend their sketch, and then grabbing the handle to change the color, width, and pattern of the trace and then continued sketching (Figure 5.16b). At the end of the session, A2 and P3 suggested using another type of interaction (e.g., voice command) to stop the robot and take over the control in urgent situations. We will consider this for future iterations of our prototype.

Application and use cases. Overall, our device will improve creativity, according to the artists, and productivity, according to the engineers. Participants also suggested various use cases for the device, such as education, architecture (e.g., drawing floor plans), textile design, rapid prototyping (e.g., website wireframe), creating floor signs for temporary events, and generating navigation patterns for other robots.

5.5 Discussion and Limitations

Below, we summarize the limitations of our current implementation and identify opportunities for future work.

Position tracking and precision. In the current setup, we use magnetic encoders to measure the distance traveled by the robot, which provides relative positioning information and is not reliable on uneven surfaces. In the future, we plan to investigate alternative techniques for position tracking (e.g., using a camera system such as the OptiTrack) that allow for absolute positioning on a wider range of surface geometries. Improving the position tracking would also allow us to sketch more complex shapes, for instance, a large raster graphic that is printed in adjacent strips. In addition, improving position tracking helps increase sketching precision, which is a major issue with plotter robots.

Form factor. The size and form factor of our robot are constrained by the size of the handheld printer. Therefore, our robot occludes part of the design during sketching. Advances in printer technology would allow us to reduce the size of the device so that it is closer to the size of physical brushes. A simple alternative would be to place a second camera underneath the case and visualize the live camera view on the display.

Manual sketching. Currently, we use a commercial dual-axis analog joystick and relative mapping to control the robot in manual mode. In future work, we plan to investigate alternative input techniques that use absolute position mapping, which would more closely resemble painting with a brush. We plan to include an omnidirectional platform with Mecanum wheels [40; 81] and backdrivable mechanism, so the joystick can be replaced by a fixed brush handle. While controlling the joystick limits manual sketching to wrist movement, replacing the joystick with a fixed brush handle would also allow movement of the entire arm. Future work should also consider integrating haptic feedback directly on the handle.

Interaction. In our current implementation, our robot immediately transitions from autonomous to manual mode when the user grabs the handle. While this approach provides convenience when the robot is in close proximity, alternative methods of interaction, such as voice commands and mid-air gestures, are being considered to address scenarios when the robot is not easily accessible. To help predict the transition time from autonomous to manual mode, in future iterations we will visualize the robot's position relative to printed marks on the device screen and backend interface.

Currently, the speed of the robot in manual mode is adjusted using the joystick. We are considering other types of interaction such as voice commands and mid-air gestures to adjust the speed in autonomous mode.

While we did not observe a split of attention between sketching and viewing the device screen during the user study, we are considering in-situ projection on the canvas to further improve the interaction with the device.

Collaborative control of the robot. Multiple users can also collaborate to control the robot. Examples include crowd participation in the creation of artwork or remote control of the robot by multiple users. This is an interesting aspect we are considering for follow-up work that opens up exciting research questions, e.g., defining the type of interaction and modality, ownership, the priority of received input, and resolving input conflicts.

Different fabrication tools. Our robot is equipped with a printer for sketching, however, it is possible to change the design of the robot and develop a modular fabrication tool. For example, the printer can be replaced with a marker, a cutter, a miniature laser engraver, or a miniature iron for sintering conductive traces. This will not only enlarge the set of fabrication tasks that can be accomplished using "handheld tools unleashed". It will also open up possibilities for new autonomous fabrication devices that collaborate with each other to accomplish a task (e.g., one robot draws a design on a fabric and the second follows the traces and cuts out the fabric).

5.6 Conclusion

So far, personal fabrication has mostly centered around handheld tools as an embodied extension of the user, or digital fabrication machines automating parts of the fabrication process without much direct user intervention. In this chapter, we explored *Mixed-Initiative Fabrication* for sketching as a continuum ranging from *manual* via *assisted* to *autonomous* fabrication, that enables seamless transitions between each mode during fabrication. As a first example of this vision, we presented *RoboSketch*, a robotic printer on wheels capable of creating large-scale, high-resolution prints. With a joystick controller, *RoboSketch* can be used for manual sketching. It also provides interactive assistance during sketching, and it can turn into an autonomous robotic device moving about for computer-generated sketches. We introduced a set of easy-to-learn interaction techniques to seamlessly transition between all three modes, along with sketching techniques that benefit from flexible transitions, e.g., to extend or revisit a sketch. Our results show that *RoboSketch*'s concept was positively received by artists and engineers, and that mixed-initiative physical sketching succeeds in making computer-supported sketching more versatile and flexible.

With this, we contributed three fabrication techniques that enable the physical sketching of circuits and sensors on diverse materials and geometries. The subsequent chapter thus explores a DIY touch-sensing technique to allow high-resolution multi-touch sensing on everyday surfaces.

6 Multi-Touch Kit: DIY Input Technique for Sensate Surfaces

The utilization of physical sketching tools and techniques, as elaborated upon in Chapters 3 through 5, has demonstrated promising results in the creation of touch sensors on various surfaces. Nonetheless, a significant challenge arises in enabling multi-touch sensing on these sensors without prior electronic expertise. Current commercial multi-touch controllers typically necessitate complex firmware programming or low-level USB programming to access raw data, thus limiting customization in terms of electrode number and size, available data, materials, and shapes of the sensor. Moreover, while some multi-touch chips are affordable, the use of these chips in a prototype usually requires the purchase of a costly development kit or the design of a breakout board and a programmer. Consequently, mutual-capacitance multi-touch sensing is primarily limited to industrial solutions or research labs with significant electrical engineering (EE) expertise, rendering this technology inaccessible to typical interaction designers and makers seeking to prototype novel touch-based interfaces.

In contrast, do-it-yourself electronic kits, such as Arduino, and their extensions have enabled non-experts to rapidly build functional electronic prototypes. One example is the Capacitive Sensing Library[1], which provides a simple firmware library to realize basic capacitive sensing using an Arduino without any specialized hardware. However, firmware is restricted to loading-mode sensing, a comparably simple mode of capacitive sensing not well suited to support multi-touch sensing grids. Our main objective is to contribute a solution of similar ease and simplicity that supports the considerably more complex

[1]https://playground.arduino.cc/Main/CapacitiveSensor

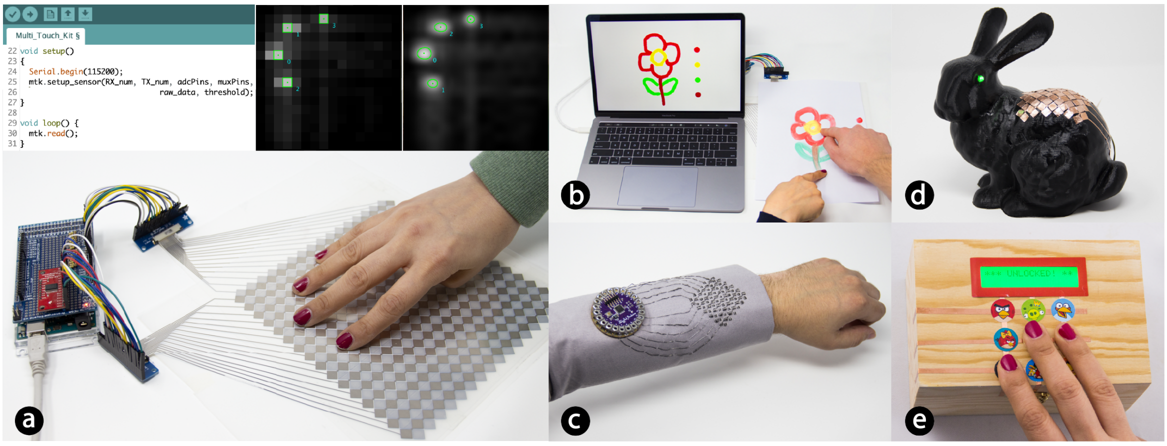


Figure 6.1 *Multi-Touch Kit* enables electronics novices to easily implement high-resolution capacitive multi-touch sensing using a commodity microcontroller (a). This supports rapid prototyping of multi-touch surfaces that are customized in dimensions, shape, and materials, for applications such as paper-based interaction (b), textile multi-touch sensing with a Lilypad (c), multi-touch input on 3D printed objects (d) and everyday objects (e).

mutual-capacitance sensing technique while avoiding the use of specialized hardware and proprietary software that thus far is required for this mode of capacitive sensing.

In this chapter, we introduce *Multi-Touch Kit*[2], a low-cost do-it-yourself technique to enable interaction designers, makers, and electronics novices alike to rapidly create and experiment with high-resolution multi-touch sensors of custom sizes, geometries, and materials.

In contrast to existing solutions, *Multi-Touch Kit* is the first technique that works with a commodity microcontroller (our implementation uses a standard Arduino) and does not require any specialized hardware. As a technical enabler, we contribute a modified multi-touch sensing scheme that leverages the human body as a transmission channel of MHz range signals through a capacitive near-field coupling mechanism. This leads to a clean signal that can be readily processed with the Arduino's built-in analog-to-digital converter, resulting in a sensing accuracy comparable to industrial multi-touch controllers. Only a standard multiplexer and resistors are required alongside the Arduino to drive and read out a touch sensor matrix.

The technique is versatile and compatible with many types of multi-touch sensor matrices, including flexible sensor films on paper or PET, sensors on textiles, and sensors on 3D printed objects. Furthermore, the technique is compatible with sensors of various

[2]This chapter is based on a publication presented at UIST'19 [165]. As the primary author, I led the development of sensing technique, hardware and software implementations, designed and fabricated the prototypes, and conducted the evaluations.

scale, curvature, and electrode materials (silver, copper, conductive yarn) fabricated using conductive printing, hand-drawing with a conductive pen, cutting, or stitching.

A comprehensive firmware and software library implements the sensing scheme, enabling developers to easily read out raw capacitance images as well as tracked locations of touch points at a high frame rate.[3]

We present empirical evaluation results that demonstrate the technique's ability to accurately detect touch input for sensors of various sizes, materials, and curvatures down to a radius of 15 mm. To verify the practical usefulness of the technique, we used *Multi-Touch Kit* to implement five technical demonstrators comprising, among others, multi-touch sensors on paper-based interfaces, 3D printed objects and textiles. These applications demonstrate the kit's ability to support high-resolution multi-touch input on sensors of up to 175×175 mm size, on flat and curved geometries of various materials. The applications further show *Multi-Touch Kit*'s compatibility with different rapid prototyping techniques, ranging from low-fidelity sketching with simple copper tape or a conductive pen to high-fidelity printed sensors.

In summary, the main contributions of this chapter are:

1. Multi-touch sensing using a commodity microcontroller without any special hardware, based on a modified multi-touch sensing approach utilizing frequencies in the range of tens of MHz for body channel transmission through capacitive near-field coupling.
2. A comprehensive firmware and software library for Arduino and Processing that enables electronic novices to easily control and read out multi-touch sensors with a few lines of code. The library gives real-time access to raw capacitance images and tracked touch locations.
3. Empirical results demonstrating accurate multi-touch sensing for sensors of various scales, materials, and curvatures.
4. Demonstration of practical usefulness with five implemented application examples.

6.1 The Multi-Touch Kit

We introduce a sensing scheme that makes it possible to sense multi-touch on a touch sensing matrix using a commodity microcontroller without any specialized hardware. We then present the do-it-yourself hardware implementation and the Arduino and Processing

[3] https://hci.cs.uni-saarland.de/multi-touch-kit/

libraries. Together, they enable novices to rapidly prototype custom multi-touch sensors and to access raw capacitance data or high-level multi-touch coordinates using a few lines of code.

6.1.1 Sensing Approach

We propose a modified multi-touch sensing approach that utilizes the extra-body transmission through an electric field. Specifically, we leverage the fact that in the frequency ranges from 100 kHz to 40 MHz the electric field around the body behaves as a quasi-static near-field [9; 220]. In our sensing scheme, we use *projected capacitive sensing* [59] with a modified *transmit+receive mode* [63; 151]: a TX electrode transmits a signal in the MHz range; in this frequency range, the quasi-static electric field allows for strong capacitive coupling between the TX electrode, the finger, and the RX electrode [220]. Simply put, the finger can be considered a conductor that couples both electrodes [9]. Since the propagation of electrical signals in the selected frequency range (< 40 MHz) is better along the human body than through the air, a touch event yields an increment in the amplitude of the received signal. This increment is significant enough to be captured by a commodity microcontroller, with a sufficient SNR for robust touch sensing. Note that this is contrasting to decrements observed in classical *shunt-mode* mCap approaches [63].

To leverage this basic principle in a touch sensor implementation, we address three aspects: (1) Investigating the *frequency response* of the touch system to select a suitable frequency, (2) *generating the effective frequency band* as a transmit signal with a commodity microcontroller, and (3) accurately *capturing the changes in the received signal* at touch events using its built-in analog-to-digital converter (ADC). We will now discuss each of these aspects and confirm their validity.

Frequency Response of the Touch Sensor

To systematically investigate extra-body transmission in the context of matrix-type touch sensing, we conducted an empirical study to derive the frequency response of the touch system in the frequency band of interest ($< 40MHz$). In this experiment, sinusoidal signals with frequencies ranging from 1 kHz to 30 MHz[4] were produced with a function generator (Keysight 33600A) and used as the transmit signal to an 8×8 multi-touch sensor (6×6 mm diamond size and 0.5 mm distance between electrodes [127]). Received signals were measured with an oscilloscope (PICOSCOPE 6402A) under two conditions: (1) when a finger is touching the sensor intersection (Touched) and (2) when there is no touch (Not

[4] 1 kHz, 10 kHz, 100 kHz, 1 MHz, and 2...30 MHz in 2 MHz intervals, all 5 Vpp

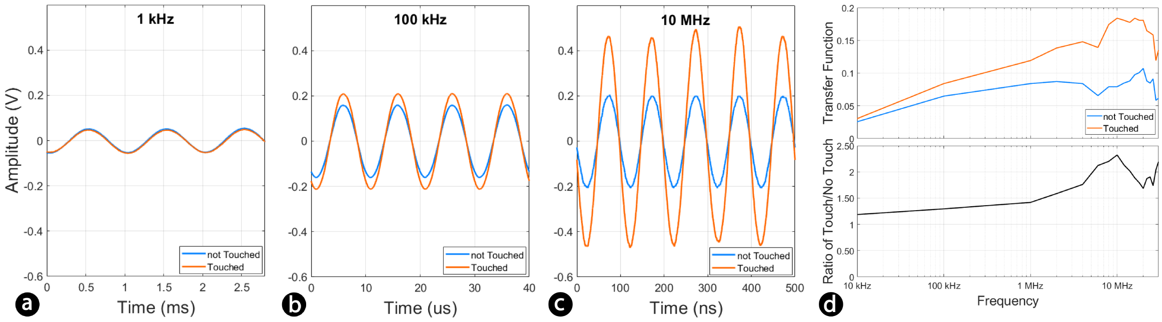


Figure 6.2 (a-b) At lower frequencies, the received signal does not show considerable change in amplitude between touched and not touched states. (c) At higher frequencies (10MHz), the extra-body propagation causes a significant increase in the received signal during touch. (d) Frequency response of the matrix multi-touch sensor for touched and not touched states. The ratio between touched and not touched states is largest at 10 MHz.

Touched). To illustrate the strong effect of frequency, Figure 6.2-a, b, and c shows the transmit and receive signals for the two conditions at frequencies of 1 kHz, 100 kHz, and 10 MHz respectively.

The amplitude of the frequency response of the touch sensor is formulated by calculating the gain or the input-to-output ratio (i.e., receive signal / transmit signal) for each sinusoidal transmit signal. Figure 6.2-d shows the frequency response of the touch sensor at conditions Touched and Not Touched for frequencies between 10 kHz and 30 MHz. An additional plot of the ratio between Touched to Not Touched is added since this is indicative of the signal-to-noise ratio (SNR): the higher the ratio, the higher the expected SNR. Therefore, the peak of the ratio curve helps us to identify the optimal frequencies for the touch sensor. As shown in Figure 6.2-d, the difference peaks at 10 MHz implying that a transmit frequency centered around 10 MHz is the optimal choice.

Generate the Effective Transmit Signal

Generating a sinusoidal signal with a commodity microcontroller is difficult since most models do not feature a digital-to-analog converter (DAC) and hence are limited to digital output. With the fixed clock frequencies in these devices, generating a specific frequency corresponding to the peak response of the body is even more challenging. As a solution to this problem, we propose to use carefully selected patterns of periodic digital signals to generate outputs with concentrated spectral power at the target frequency band. Most commodity microcontrollers are capable of creating a wide range of pulse width modulation (PWM) signals, which are digital square waves with different duty cycles. Using Fourier analysis, these PWM signals can be represented as a collection of sinusoidal signals

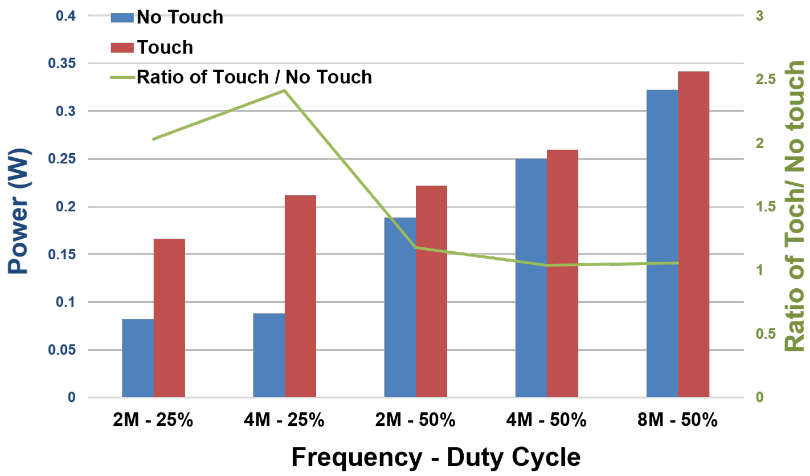


Figure 6.3 Total energy of received signals in different frequencies and duty cycles. The greatest difference between the energy of touch and no touch signals belongs to the 4 MHz frequency with a 25% duty cycle, which makes it a suitable choice for touch detection.

(harmonics) spread across a wide bandwidth. Our approach is to identify a PWM signal (both frequency and duty cycle) that has high amplitude harmonics in the peak areas of the frequency response of our multi-touch sensor.

To identify the suitable frequency and duty cycle for a PWM signal with harmonics in the optimum frequency band, we conducted an empirical study by recording the received signal (using the oscilloscope) for a set of transmit PWM signals. As inputs, we selected $2MHz$, $4MHz$ and $8MHz$ PWMs with 25% and 50% duty cycles. We selected these configurations to represent the available PWM frequencies in Arduinos and to spread the power in both odd (25%) and even harmonics (50%).[5] The output power is calculated based on Parsevals Theorem by (1) deriving the Fourier series of each received signal to represent it as a summation of individual sinusoidal harmonics, (2) calculating the sum of squares of each coefficient divided by two, and (3) then adding the square of the DC component. We derived the output power for each input PWM configuration for the two different conditions (touched and not-touched as shown in Figure 6.3. Results show that the $4MHz$ signal with 25% duty cycle outperforms the other PWM configurations. For illustration, Figure 6.4 shows the prominent harmonics of this PWM signal. They are close to our target transmit frequency of 10 MHz.

[5] $8MHz$-25% was not used since Arduino Uno and Mega cannot generate this signal.

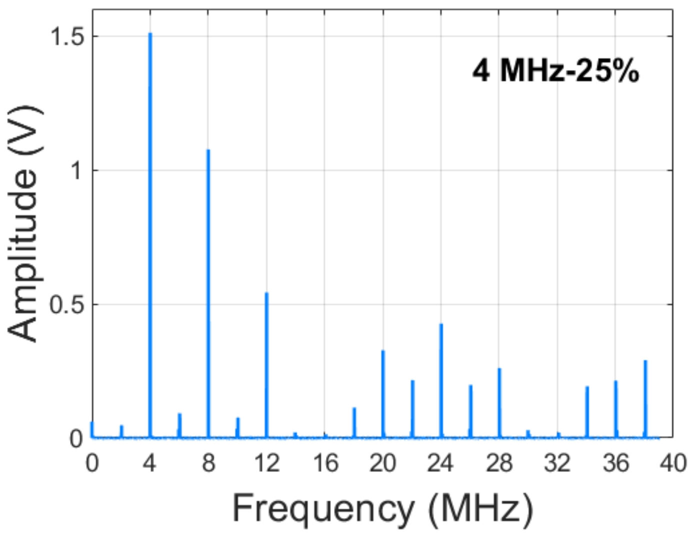


Figure 6.4 The prominent harmonics of 4MHz, 25% duty cycle are close to 10 MHz.

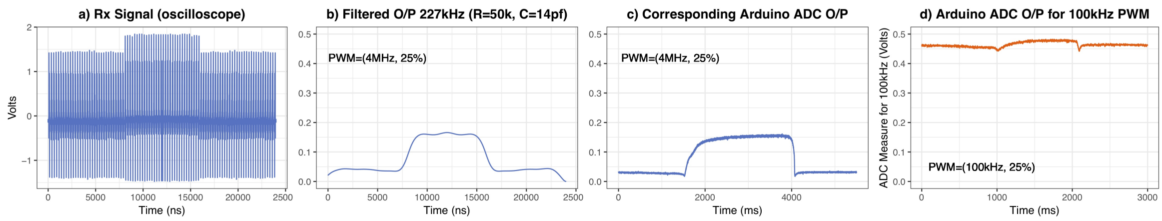


Figure 6.5 (a) Receive (RX) signal of the sensor showing no-touch to touch and back to no-touch transition (Transmit signal $4MHz$, 25% duty cycle); (b) RX signal filtered with Arduino internal ADC RC configuration (average R = 50k, c=14pf); (c) Output of the Arduino's ADC; (d) In contrast, a low frequency (100KHz, 25% square wave) transmit signal does not allow for robust touch sensing using the ADC.

Analyzing Receive Signal

Most commodity microcontrollers lack the capability to accurately sample a signal of this high frequency. For instance, the highest sampling frequency with accurate ADC conversion for ATmega328P-based microcontrollers (most Arduinos) is limited to $1MHz$.

To overcome this challenge, we leverage the fact that human touch-down and touch-up events occur at a much lower frequency than the actual PWM frequency of TX and RX signals. This results in an amplitude-modulated signal where the increase in amplitude due to touch contact envelopes the PWM signal as shown in Figure 6.5-a (captured with Picoscope, USB oscilloscope). Amplitude-modulated signals can be easily recovered using a simple low-pass filter (LPF).

The internal architecture of the Arduino analog-to-digital (ADC) converter implements a low-pass filter. The ADC utilizes a sample-and-hold capacitor ($14pf$). This capacitor, along with the path resistance (ranging from $1k\Omega$ to $100k\Omega$[6]), creates an internal low-pass filter (LPF). The cut-off frequency f_c of this filter is well below the PWM frequency. Assuming a resistance value in the center of the range specified in the data sheet ($R = 50k\Omega$), with $C = 14pf$, the cut-off frequency is $f_c = \frac{1}{2\pi RC} = \frac{1}{2\pi \times 50 \times 10^3 \times 14 \times 10^{-12}} = 227.4kHz$. This LPF filters the high-frequency components of the received signal, leaving the attenuated low-frequency touch signal to be converted as the ADC values.

To demonstrate the effect of this low-pass filter, we modeled it with *R*'s *Signals* package and applied it to the captured raw received signal. Figure 6.5-b shows the signal after applying the low-pass filter. It shows that the filtered signal accurately represents the touch state. Figure 6.5-c shows the microcontroller's ADC output (converted to Volts) for the transition from no-touch to touch to no-touch. It shows the values corresponding to the low-pass filtered signal. For comparison, we also captured the ADC output with a lower frequency PWM signal ($100kHz$, 25% square wave). As shown in Figure 6.5-d, this results in a poor SNR, demonstrating the greatly superior performance of the high-frequency signal.

6.1.2 Hardware Implementation

Considering its popularity among the HCI and maker communities, we selected the Arduino platform as our foundation hardware unit. *Multi-Touch Kit* limits the use of external hardware to a commonly available simple multiplexer and standard resistors. It is compatible with a variety of multi-touch sensor matrices to support versatile prototyping.

Hardware Components and Interconnection

Multi-Touch Kit hardware schematic is shown in Figure 6.6. We tested setups with Arduino Uno (ATmega 328P), Arduino Mega (ATmega 2560), and Arduino LilyPad (ATmega 328P), all very popular microcontrollers. Arduino's hardware *Timer 2*, the internal crystal oscillator of the controller, is used as the clock generator to generate a 4 MHz square wave with 25% duty cycle of 5 Vpp magnitude via pulse width modulation (PWM). Since the high-frequency PWM signal is limited to a few pins, we use a standard multiplexer (CD74HC4067, < $1) to drive multiple transmitter lines. This multiplexer is a general-purpose component and users can freely choose their own multiplexer since it has the

[6] ATmega328P datasheet Figure 23-8

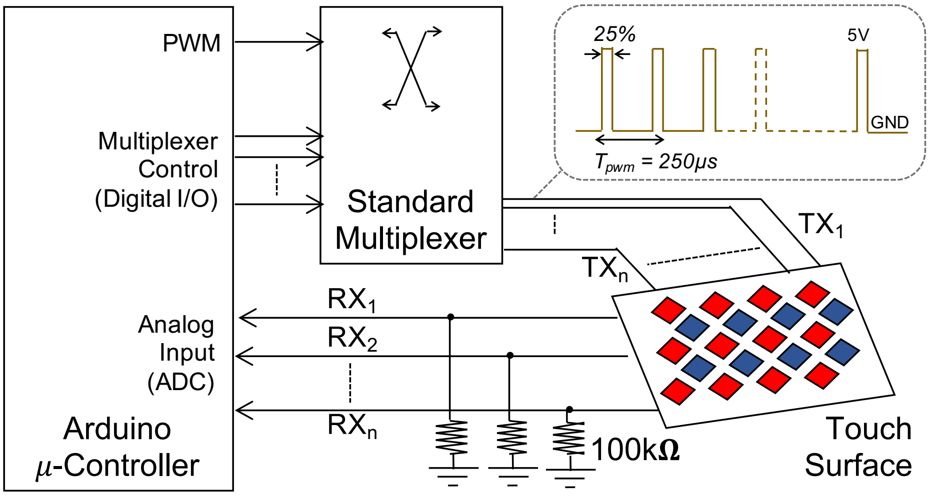


Figure 6.6 *Multi-Touch Kit* hardware schematic and interconnection

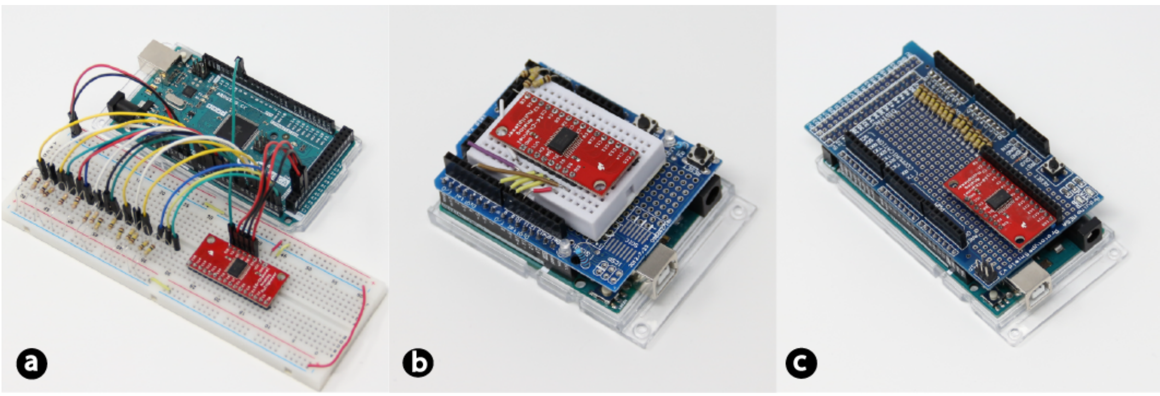


Figure 6.7 The simple hardware setup can be implemented on (a) a breadboard, (b) using an Arduino Proto Shield without soldering, and (c) with soldered connections.

bandwidth to work in the functional frequency range. The receiver terminals are connected to $100k\Omega$ load resistors. The voltage across load resistors is measured using the analog input pins of the Arduino. As shown in Figure 6.7-a, the complete setup can be easily implemented on a breadboard, even for sensor matrices of considerable size (16 × 16). Alternatively, a more compact and physically more robust setup can be realized with an Arduino Proto Shield, either using an embedded mini breadboard for solderless operation (Figure 6.7-b), or by soldering components (Fig. 6.7-c). It is also possible to realize the full setup using an Arduino Lilypad with a Lilypad prototype board, connecting the Lilypad setup to the sensor with snap-in connections for ease of use in textile applications (Figure 6.1-c).

Fabrication of Multi-Touch Sensor Matrix

Multi-Touch Kit is compatible with established rapid prototyping techniques for fabricating sensor matrices that have been presented in the literature. These include *conductive inkjet printing* on a desktop printer [98], as demonstrated in [140], or cutting *copper foil* using a commercial vinyl cutter, as used in [189]. Sensor designs can also be hand-drawn with a *conductive pen* (Circuit Scribe) or stitched on fabric with *conductive thread* (Adafruit Stainless Thin Conductive Thread). For very rapid, low-resolution designs, it is even possible to manually apply strips of copper tape in rows and columns, as we will demonstrate in the application section. Sensors can be curved down to a radius of 15 mm.

We recommend using the classical two-layered diamond pattern that is commonly used for mutual-capacitive touch sensing [42; 127], with electrode dimensions ranging between $4\times4\ mm - 6\times6\ mm$ and an inter-electrode spacing of 1 mm. Our library includes reference vector graphic designs that can be directly printed or cut.

6.1.3 Software Implementation

We provide an Arduino firmware library and a Processing library. It offers an API for easy access to low-level raw capacitance values and high-level touch coordinates while hiding the low-level logic of our sensing approach from the application developer.

Arduino Library

The Arduino library internally sets the responsible registers to configure the relevant frequency and duty cycle of the PWM signal. It further sets the reference voltage for the analog-to-digital converter and controls time-division multiplexed scanning of the sensor matrix internally. The library reports raw capacitance values. Alternatively, for rapid prototyping, it can report binary touch up/down states based on simple thresholding (more advanced touch blob analysis and tracking are offered in the Processing library).

Only two functions are required to be called in an Arduino program to interface with the sensor:

`setup_sensor()`: This function needs to be called only once in the setup() function of the Arduino program. It accepts the following arguments: sensor dimensions (the number of TX and RX lines), an array with numbers of analog-in ports connected to RX pins, an array with numbers of digital I/O pins connected to control the multiplexer, a Boolean variable defining whether raw capacitance data or touch up/down states shall be reported, and a threshold for a touchdown state.

`read()`: This function returns a two-dimensional array of 10 bit raw capacitance readings or binary touch-up/down states corresponding to each row-column intersection. Each function call completes a full scan of the touch sensor.

With an Arduino Mega and for sensors with dimensions 4×4, 8×8, and 12×12, the read function on average took $1.85ms$ ($SD = 0.45$), $7.2ms$ ($SD = 0.46$) and $16.38ms$ ($SD = 0.49$) respectively to complete. Therefore, with a 12×12 sensor, the highest achievable frame rate is 60 *fps*.

Processing Library

To convert the raw capacitance values returned by the Arduino library into high-level multi-touch information, data need to be (1) calibrated and scaled, (2) interpolated and merged, and (3) blobs extracted and touch points tracked over consecutive frames. To streamline the process, we have created a software library for the frequently used open-source prototyping platform Processing[7]. It parses raw touch data sent from the Arduino through the serial port for further processing.

Calibration and Scaling Raw data of mutual-capacitance sensors are affected by several internal and external factors of the sensor design. For instance, previous research shows the intensity of raw values varies with distance from the connecting edge of a sensor matrix [140]. Additionally, in custom designs, the custom size, length, shape, and materials used for the electrodes may also affect the homogeneity of the raw values. Therefore, raw values are first calibrated and normalized.

The calibration process is done once per sensor per user, and the results are saved. It consists of two steps. First, we remove the offset values of individual cross-section characteristics by subtracting the average noise floor. Then, the developer is asked to touch a random place on the sensor surface. The reported values are saved and used to normalize and pre-scale the data before interpolation. This process can be automated.

Interpolation and Merging Data from the sensor are organized in arrays of integers representing each TX and RX intersection. These arrays are merged into a 2D image creating a capacitive image of the sensing surface. However, because of the limited number of electrodes, this image has a low resolution. Similar to previous research in capacitive sensing, we use bi-cubic interpolation to create an up-scaled image suitable for blob detection and finger tracking using image processing techniques [140].

[7] https://processing.org/

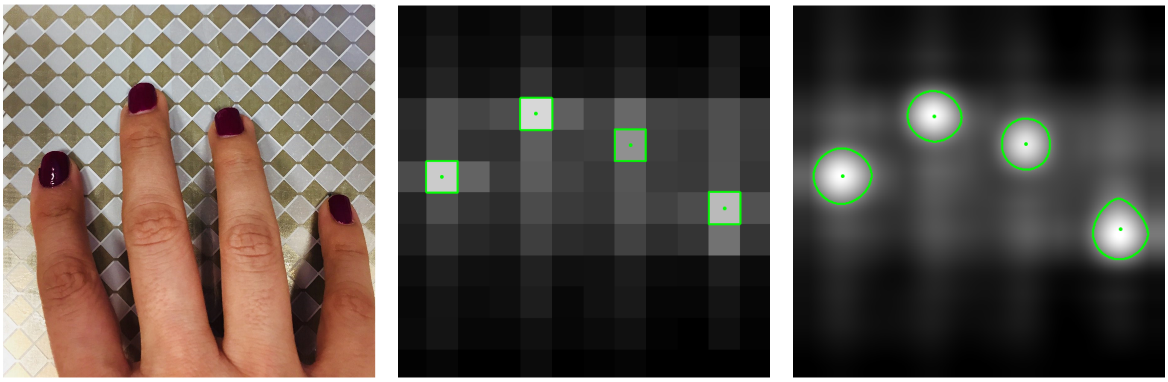


Figure 6.8 Detecting multi-touch input on a 12x12 sensor: (a) four fingers touching the sensor (b) raw capacitive image, (c) interpolated image with blob detection.

Touch Detection and Tracking Finally, to identify the touch locations on the surface of the sensor, we use the OpenCV[8] library for the blob extractions from the capacitive image. The centers of these blobs are detected as touch points (Figure 6.8). Each touch point is labeled with a unique ID and tracked over consecutive frames.

6.2 Evaluation

In order to verify the functionality of the *Multi-Touch Kit* as a prototyping platform, we conducted a series of technical studies. These evaluate the signal-to-noise ratio (SNR) of multi-touch input on sensors of various scales, the accuracy of touch location, the effect of curved geometries, and the effect of different electrode materials and fabrication approaches.

For all of our experiments, we used a fully mobile, battery-powered setup (similar to Figure 6.1a, except the data sent via Bluetooth to a PC), as this creates the most challenging grounding condition for capacitive touch sensing [62]. We used an Arduino Mega microcontroller, which sent the raw data to a PC via a Bluetooth connection. Except for evaluating different materials, all sensor samples were printed on transparent PET film using a Canon IP100 desktop inkjet printer and conductive silver ink [98]. The TX and RX electrode layers are printed on separate PET films and then attached together with a very thin layer of adhesive film. The top surface of the sensor is insulated with a thin layer of transparent dielectric. During the experiments, the sensors were placed on the surface of a wooden table.

[8]https://opencv.org/

6.2.1 Signal-to-Noise Ratio and Scalability

Different applications demand customized multi-touch surfaces of various sizes. The most important factor to support such customization is the sensor's ability to scale while offering a sufficiently high signal-to-noise ratio for robust touch sensing. We conducted a pilot study to identify the most demanding test conditions and then evaluated touch input in these conditions with 10 participants.

Pilot Study

To identify the most demanding touch conditions for the main study, we conducted a pilot study. We used a sensor with 12×12 electrodes 6×6 mm diamond shape [42; 127]. We followed a factorial design with four locations on the touch surface and four multi-touch cases to test. Corners were selected to represent the most challenging locations compared to the connecting edges [140]. For each corner, we tested four touch conditions: single-touch, simultaneously touching with a second finger on the same TX line, touching with a second finger on the same RX line, and touching with three fingers (on the corner, TX, and RX lines). In the case of multi-touch input, secondary and tertiary fingers were positioned on the respective TX or RX line at the position closest to the corner while still being detected as its own touch point. We had previously identified that this is the most demanding multi-touch condition in terms of signal-to-noise ratio. As the dependent variable, we calculated the signal-to-noise ratio (SNR) of the touch input, which is the most commonly used measure to evaluate the quality of touch sensing [36]. For each condition, 5 iterations of SNR values were recorded.

The results revealed that touching the closest corner to the connecting edge for both RX and TX lines had the lowest SNR (57, $SD = 20.0$). Furthermore, it became apparent that additional multi-touch contacts reduced the SNR compared to single-touch sensing. We further tested these touch conditions under various grounding conditions of the user (sitting with legs resting on the floor, sitting with lifting the foot, and standing on the floor) and changing the sensor position (put on the table, handheld, or put on the arm with isolation layer between the sensor and the skin). These conditions did not considerably affect the SNR. The highest change we observed was 3%. In light of the high SNRs identified in our experiments, this effect is negligible.

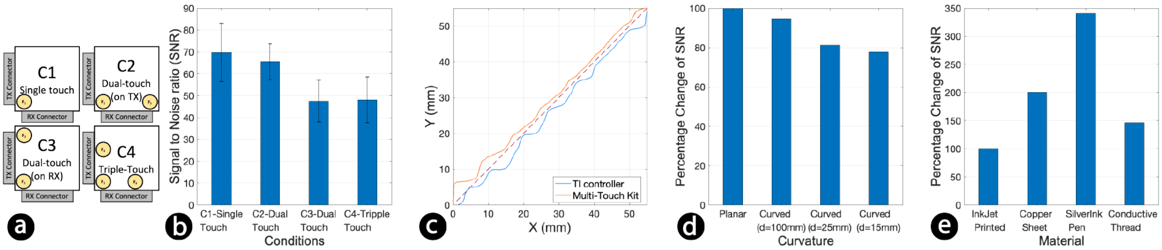


Figure 6.9 Evaluation results: (a) Locations tested in user study represent the most demanding locations, (b) Signal-to-noise ratio for touch input with 1, 2, and 3 fingers on a sensor with 12x12 electrodes, (c) Spatial accuracy of touch input; (d) Effect of sensor curvature on SNR; (e) Effect of electrode material on SNR.

User Study

We conducted a controlled experiment with users to more formally investigate the signal-to-noise ratio of our sensing approach and to account for the effect of body capacitance, which is known to vary across users. We recruited 10 participants (4 female, mean age 35).

We selected the most demanding conditions identified in the pilot: the sensor with 12×12 electrodes and 6×6 mm electrode dimensions. On this sensor, we chose the location that had performed the least well in the pilot study: the corner closest to TX and RX connectors. By showing a sufficiently high SNR in this most demanding case, we will be able to show the overall robustness of the sensing approach. The sensor was placed on a table. The participant was standing.

As conditions, the study had four different touch locations, which are indicated in Figure 6.9-a. They comprised the most demanding single-touch, dual-touch, and triple-touch locations we had identified in the pilot. In each condition, the participant was asked to touch the sensor consecutively five times with one, two, and three fingers at the respective positions that were visually marked on the sensor, for one second with a one-second pause in-between touching. Raw capacitance data were sent to a PC through a Bluetooth connection and logged for analysis. Figure 6.9-b shows the average signal-to-noise ratio for each condition and all participants. The results show that all values are well above the critical value of 15, which is required for robust touch sensing at industrial strength [36].

6.2.2 Spatial Accuracy

To measure the spatial accuracy of touch sensing and to compare it with the baseline of an industry-strength commercial touch controller, we recorded finger movement on a sensor and compared the interpolation results with ground truth. We selected the Texas

Instruments TI MSP-CAPT-FR2633 touch controller chip for the baseline comparison. Since this controller supports a maximum of 16 I/O pins, we used a 8 × 8 multi-touch sensor of 55 × 55 mm size for this experiment. Following the method presented in [6], we visually marked the main diagonal axis of the sensor starting from the electrode farthest from the signal driving lines. The diagonal axis was selected, as it is to be expected that the accuracy of interpolation is lowest because of the larger distance between electrode intersections. The finger was dragged diagonally through the sensor along the marked line. This was repeated 5 times with the sensor connected to *Multi-Touch Kit* and 5 times with the sensor connected to the commercial touch controller. The resulting raw data were recorded and used for interpolation and calculation of the touch locations.

The Root Mean Square Error (RMSE) of each trial was calculated. The average RMSE for *Multi-Touch Kit* is 1.56 mm (SD=0.17) and for the TI controller 1.94 mm (SD=0.20). The results show that our toolkit has a spatial accuracy comparable to the commercial touch controller. For qualitative visual inspection, the results of the trial with the highest RMSE in either condition are depicted in Figure 6.9-c. The plot shows that the sensed locations of *Multi-Touch Kit* closely match with the ground truth. The maximum offset is less than 3.90 mm, which is close to the natural imprecision of human touch [78].

6.2.3 Curvature

To evaluate the effect of curved sensor geometries, we conducted a technical evaluation with four conditions: planar and 3 curved geometries with a diameter of 100 mm, 25 mm, and 15 mm each. The larger diameter reflects the typical curvature of everyday objects such as mugs, while the smallest one reflects objects such as markers or pens.

The experiment was run with a 4 × 4 electrodes, 30 × 30 mm sensor. The small dimension was chosen to be able to wrap the sensor around surfaces of very small diameter. Touched and not touched events (1 s interval, 5 trials) were captured at the most demanding intersection (closest to the transition lines) and the SNR was calculated. Figure 6.9-d presents the percentage change of SNR with respect to the planar condition. As expected, the planar condition has the highest SNR, since the finger has maximum contact with the sensor surface. The SNR of the most curved condition was 22% lower. Considering the very high (well above 40) that we have identified above for the most demanding touch conditions (Figure 6.9-d), it is apparent that even a considerably larger reduction would still ensure SNR values above 15 (the required value for robust touch sensing). This demonstrates that the sensing approach is robust for curved geometries.

6.2.4 Materials

Finally, we investigated the effect of using different materials and fabrication methods for the physical sensor matrix: inkjet-printed with conductive silver ink, hand-painted with a silver pen, stitched with conductive yarn, and cut out of the copper sheet. For each condition, we fabricated a sensor with 6 × 6 electrodes and 45 × 45 mm dimension. As the insulation layer, we used transparent PET film (~ $70 \mu m$ thickness), standard A4 office paper ($80 g/m2$), embroidery fabric (Muslin, thread count of 150), and overhead PET film (~ $100 \mu m$ thickness), respectively.

All sensors were placed on a wooden table. Each sensor was touched 5 times with the index finger (1 s touched, 1 s released) at the most demanding location (closest intersection to the driving lines).

Figure 6.9-e depicts the percentage change of SNR with respect to the inkjet-printed sensor. The results show that the SNR further increases for other materials. Copper is more conductive than silver-printed electrodes. While the sheet resistance of electrodes hand-drawn on paper with a conductive pen is higher than of inkjet-printed electrodes, using paper results in a thinner dielectric layer. This fully compensates for the loss in conductivity. The textile solution with conductive thread, in turn, benefits from having the transmitter and receiver electrodes on the same side of the textile substrate. Overall, these results confirm the compatibility of our sensing approach with different materials and their fabrication techniques.

6.3 Example applications

In order to showcase the functionality and versatility of *Multi-Touch Kit*, we developed five applications. These applications demonstrate the use of the toolkit with different materials, substrates, scales, geometries, and fabrication methods. They span from rapid physical prototyping utilizing copper tape to high-fidelity and high-resolution printed, hand-drawn, or embroidered sensors. To fabricate high-resolution sensors, *Print-A-Sketch* or *RoboSketch* handheld tool can be employed, and for more precise hand-drawn sensors, *BodyStylus* handheld tool in combination with projection mapping can be utilized (see Figure 6.10).

6.3.1 High-Resolution Interactive Surface

To turn surfaces in the physical environment into a high-resolution input surface, we designed a customized multi-touch sensor with a 16×16 electrode matrix of 177×177 mm

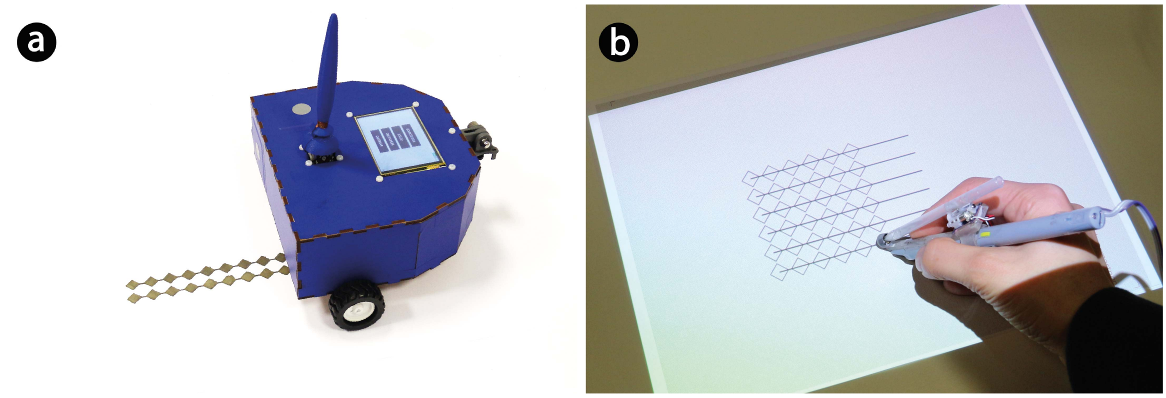


Figure 6.10 Fabrication of high-resolution sensate surfaces: (a) printed sensor with *RoboSketch*, (b) hand-drawn sensor using *BodyStylus* system.

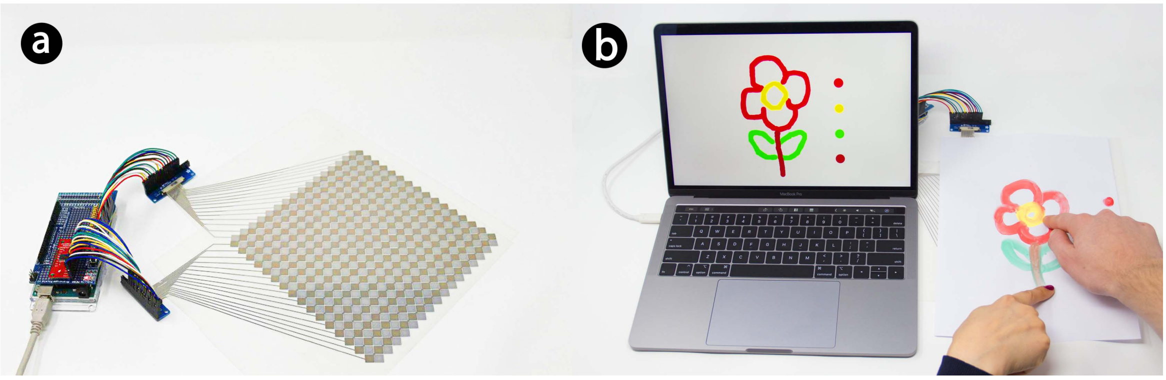


Figure 6.11 High-resolution interactive surface: (a) 16x16 multi-touch sensor is connected to an Arduino Mega, (b) digital capture of finger painting is visualized on a laptop.

size. Electrodes were printed on a desktop inkjet printer (Canon PIXMA iP100) using silver ink (Mitsubishi NBSIJ-MU01). The sensor was tethered to an Arduino Mega with an extension board containing the multiplexer (Figure 6.11-a). The sensor supports multi-touch input of up to 10 fingers and can be used for various high-resolution and multi-touch scenarios.

As one example, we implemented an interactive finger painting application. The application uses the Arduino and Processing library to directly retrieve tracked touch coordinates. A sheet of office paper is placed on top of the sensor. The user can then create a colorful physical drawing using colors and drawing with one or multiple fingers simultaneously. A high-resolution digital copy of the painting is captured by the touch sensor and visualized in a viewer application that runs on a laptop (Figure 6.11-b). The color of digitally captured strokes can be set in the application. To draw a new painting, the user only needs to replace the paper while keeping the sensor sheet.

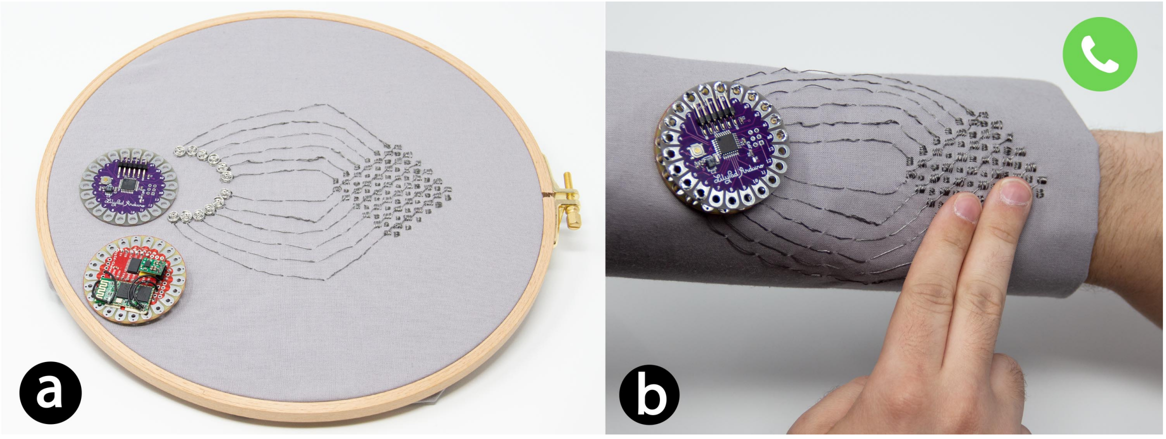


Figure 6.12 Textile multi-touch sensor with conductive yarn: (a) Sensor setup with Arduino Lilypad and Proto Board. (b) Tapping the sensor with two fingers accepts a phone call.

6.3.2 Textile Multi-Touch Sensor with Conductive Yarn

To demonstrate that *Multi-Touch Kit* supports sensors on various materials and substrates, we created a textile multi-touch sensor (Figure 6.12-a). It contains a 6 × 6 grid of diamond-shaped electrodes that were stitched on a textile using conductive yarn (Adafruit Stainless Thin Conductive Thread). While a programmable sewing machine could have been used for this purpose, we opted for stitching by hand to confirm the functionality even for the less accurate manual fabrication approach. After stitching, we used coating spray (Kontakt Chemie 74313-AA) to isolate the transmitter line.

For textile compatibility, we used an Arduino Lilypad and a Lilypad prototype board containing the multiplexer, Bluetooth and battery. The Lilypad was connected to the textile sensor with snaps, which makes it easy to attach and detach from the garment. The setup was connected via Bluetooth to a smartphone that recognizes simple gestures.

Inspired by [161], we embedded our sensor on the sleeve of a shirt to offer direct interaction with a mobile device while the user is on the go (Figure 6.12-b). Swiping with three fingers to the right or left is mapped to switching between music tracks; tapping with two fingers accepts an incoming call, while covering the sensor with the full hand rejects the call.

6.3.3 Multi-Touch Sensor on 3D Printed Object

Our technique is compatible with curved multi-touch sensors on 3D-printed objects. We 3D printed a Stanford bunny on an FDM printer (Ultimaker S5) and turned it into a multi-touch sensitive interactive object (Figures 6.13 & 6.1-d). We covered the rabbit's

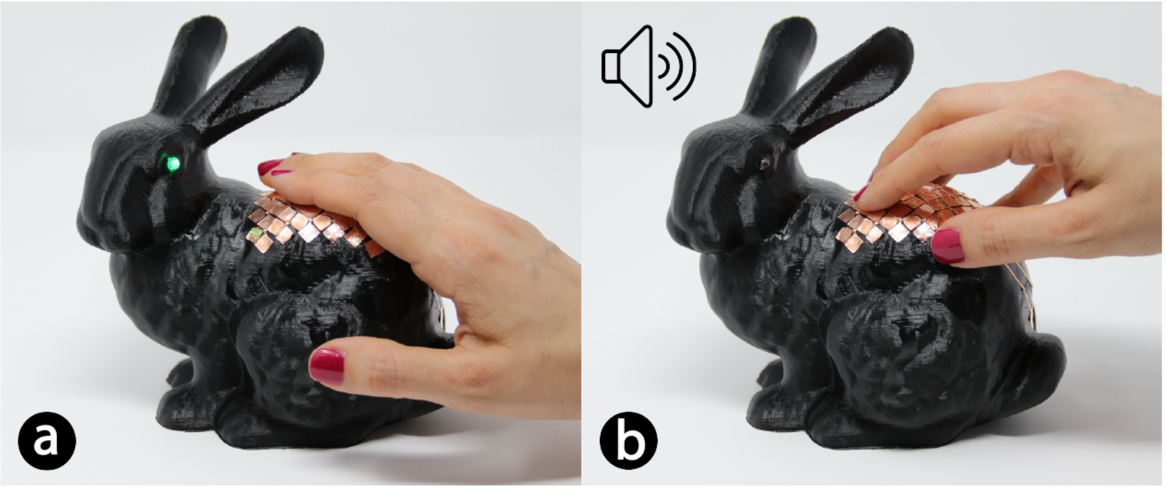


Figure 6.13 Multi-Touch sensor on 3D printed object: (a) cuddling the rabbit's back lights up the rabbit's eyes, (b) poking causes a purring sound.

curved back with a 6 × 6 multi-touch sensor. The sensor was made from a copper sheet cut with a vinyl cutter. The transmitter and receiver layers were mutually isolated with a transparent, acrylic coating spray (Kontakt Chemie 74313-AA). We added LEDs to the rabbit's eyes and a speaker to the body. When the user cuddles the rabbit's back with the full hand, the rabbit's eyes light up. When poking it with a finger, the rabbit makes a purring sound (Figure 6.13-a,b).

6.3.4 Interactive Greeting Card with Hand-Drawn Sensor

To demonstrate that our kit works with sensors that are hand-fabricated using a conductive pen, we realized an interactive greeting card that can play music and is controlled using touch input. The greeting card contains a color image that was printed using a color desktop printer (Figure 6.14-a,b). We used a conductive pen (Circuit Scribe) to draw a 4 × 4 multi-touch sensor pattern alongside conductive lines to connect the sensor and surface-mount LEDS that we attached to the card (Figure 6.14-c). The card was connected to an Arduino Uno with a Bulldog clip containing wires (Figure 6.14-d). We mapped swiping right or left to "turn on and off the LEDs" and tapping with two fingers to "turn on and off the music".

6.3.5 Rapid Prototyping with Copper Tape

The early phases of a design process commonly involve quickly exploring a large number of design alternatives at low fidelity. Implementation time is critical here, as it would be

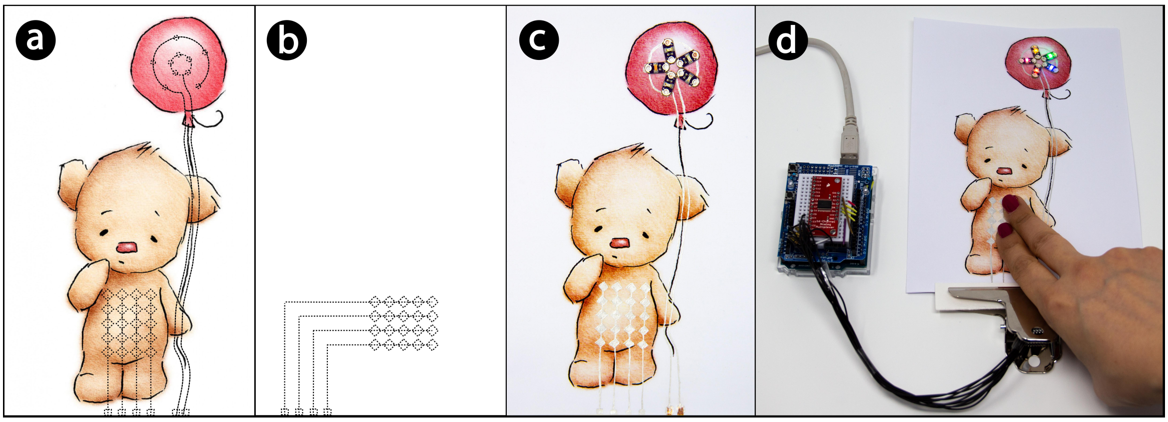


Figure 6.14 Interactive greeting card: (a) Front side of the greeting card with dot patterns highlighting the sensor area, (b) backside of the card, (c) fabricated card contains hand-drawn multi-touch sensor and LEDs, (d) the sensor is connected to an Arduino Uno with a clip.

prohibitive to invest significant time for every design alternative. We demonstrate the use of *Multi-Touch Kit* for very rapid prototyping with a handmade low-fidelity sensor made of simple copper tape. The sensor is fabricated within a minute by applying strips of copper tape to the desired input location to form a matrix of rows and columns and then connected to an Arduino Uno, which is placed inside the box (Figure 6.15-a). While the effective sensing resolution is certainly lower than in our other demonstrators, as the strips do not form a dense diamond pattern, it can be sufficient in many cases of low-fidelity prototyping. For this application, we used the touch event detection of the Arduino library.

We demonstrate this with an interactive wooden treasure box (Figure 6.15-b). The box can be unlocked by simultaneously touching a secret combination of locations on the sensor using multiple fingers. We attached 6 strips of copper tape on the lid of the box to create a 3 × 3 touch sensing matrix. To help the user remember the correct locations, we added different graphical icons on top of each intersection. The box is unlocked only if the correct combination of three images is touched.

6.4 Limitations and Future Work

Results from the technical evaluation and the successful implementation of the applications show that *Multi-Touch Kit* can accurately detect multi-touch input with sensor matrices of different scales, curvature, and materials. We experimentally validated its functionality for sensors up to 12 × 12 electrodes; anecdotally we can confirm its functionality for 16 × 16 electrodes, as we have used this larger size for the high-resolution

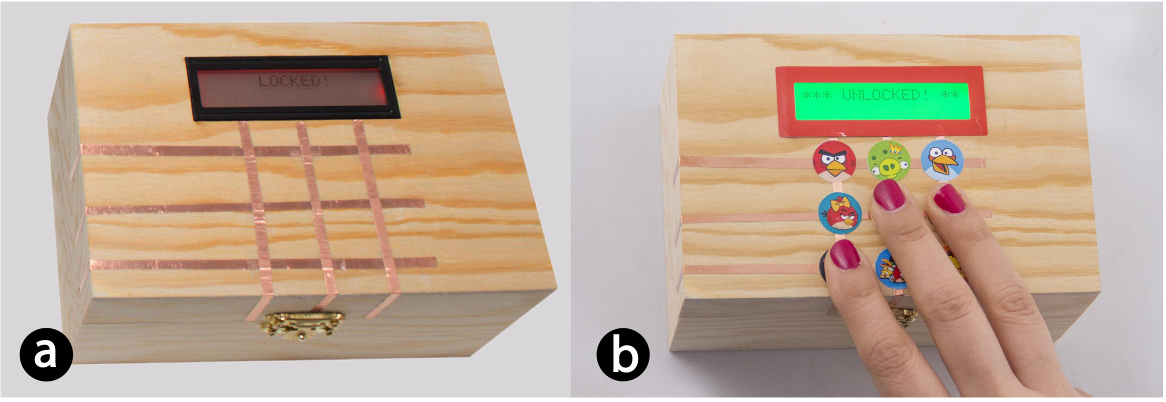


Figure 6.15 Wooden treasure box: (a) Rapid prototyping with copper tape on a wooden box, (b) touching three different images unlocks the box.

interactive surface used in the first example application. Visual evidence of the sensing accuracy of this large sensor size is provided in the companion video. However, compared to commercial multi-touch controllers, our rapid prototyping approach is subject to several limitations:

Since the sensing approach is based on extra-body propagation of signals, it is not possible to capture input made with conductive objects. Furthermore, due to high-frequency signals, the approach is less well suited for sensors made of high-resistance materials, such as ITO. We also observed that the properties of the dielectric materials used on the top and between sensing electrodes have a significant effect on the SNR. We observed that thicker top insulators (more than 400 μm) will render the interface unusable. This also implies that the hover state is not captured by the sensor.

We never experienced any issues of RF interference during the development and use of our sensors. We have further tested the interference of our sensor on nearby devices with an AM/FM radio and could not detect any noise. To test the operation of the sensor when integrated into other electronics, the sensor was placed close to an active LCD display and main power cable. We did not observe any effect on the sensor reading, nor on the operation of the other electronic devices. Our approach is compliant with the FCC regulations on equipment authorization of home-built radio frequency devices [1].

Our current prototypes are implemented with the Atmel megaAVR family of micro-controllers. Due to variations in the internal ADC and PWM implementations, other micro-controllers may have different responses. In future work, we plan to fabricate and test larger sensors and extend the hardware support for our open-source library by including other frequently used commodity platforms such as Teensy and Raspberry PI.

6.5 Conclusion

In this chapter, we contributed a technique for the do-it-yourself prototyping of capacitive multi-touch sensors. By utilizing the improved extra-body propagation of electrical signals at higher frequencies, we demonstrated the feasibility of implementing capacitive multi-touch sensing on commodity microcontrollers without the need for specialized hardware. The technique proposed here, along with the Arduino firmware and Processing libraries provided, enables the realization of custom applications that incorporate capacitive multi-touch sensing for a broad range of users, including DIY enthusiasts, interaction designers, and students. The results of our technical evaluation revealed a high signal-to-noise ratio and high spatial accuracy, thus providing robust multi-touch sensing capabilities for interactive prototypes. Additionally, our approach is compatible with sensors of various scales and curvatures, with formal evaluations conducted on sizes up to 12x12 electrodes, and informal testing showing support for larger sizes of up to 16x16. Furthermore, the results of our technical studies and implemented application demonstrations indicate that the technique is compatible with sensors fabricated using multiple materials and various rapid prototyping techniques.

7 Conclusions

Sensate surfaces, as a subcategory of interactive surfaces, are characterized by their ability to detect various forms of input, including touch, pressure, temperature, and motion. This enhances the user experience through direct manipulation of objects, making interaction more intuitive. The fabrication of sensate surfaces involves the integration of sensors into the surface, which can be achieved through various techniques such as screen printing, inkjet printing, and embroidery. The resulting sensors are then connected to a microcontroller or other electronic components for processing and interpreting input, as well as controlling output. However, current prototyping approaches have a number of limitations, including a separation of the design and fabrication process from the final object, and the need for advanced knowledge of the underlying technology and specialized hardware to prototype multi-touch sensors, which are critical for rich interaction on interactive surfaces.

The aim of this thesis was to address these challenges and contribute to the fields of interactive fabrication, rapid prototyping, and ubiquitous computing. In this final chapter, the main contributions of this thesis are summarized, the three challenges initially introduced in the introduction are revisited, and an outlook on potential directions and challenges for future work is provided.

7.1 Summary

This thesis advances the field of physical sketching of sensate surfaces in the following points:

Physical Sketching Tools and Techniques

The contributions of this thesis present novel tools and techniques for the fabrication of customized sensate surfaces. In Chapter 3, the *BodyStylus* tool and technique were introduced. Inspired by body-art practices, it merges the ease and expressiveness of freehand sketching with digital assistance. A collection of design and fabrication techniques were outlined, incorporating in-situ projected guidance and physical constraints, to simplify the creation of functional and aesthetically pleasing circuits and sensors on diverse surfaces such as the human body. The *BodyStylus* handheld tool features a custom-designed dispensing pen and self-sintering conductive ink to instantly create functional traces.

In order to extend the fabrication capabilities beyond the skin and cover everyday objects while combining the benefits of manual sketching with high-resolution printing, Chapter 4 introduced *Print-A-Sketch*. The *Print-A-Sketch* handheld tool comprises a high-resolution inkjet printer, an optical flow sensor, and an RGB camera for sensing and tracking. *Print-A-Sketch* techniques address three main challenges of physical sketching with a handheld printer: compensating for variations in hand movements during sketching, aligning the current design with previously printed patterns, and adjusting ink dispensing to the properties of the material to create functional interfaces.

In Chapter 5, the range of sketching is expanded beyond the reach of the user's hand and arm movement, and the scope of collaboration between humans and machines in the creative design and fabrication processes is broadened. The *RoboSketch* artifact, which comprises a robot on wheels with a high-resolution color inkjet printhead, can be used as a handheld tool for manual sketching, supports assisted sketching, and acts as a robotic partner for autonomous drawing. *RoboSketch* mixed initiative techniques allow close collaboration between humans and machines in the creative design and fabrication process. The sketching techniques support designers, makers, and artists in sketching out their initial ideas, iteratively extending them, and revisiting the composition to complete details.

Touch Sensing Techniques

The present thesis also contributed to the advancement of customized high-resolution multi-touch sensors on everyday surfaces. The physical sketching tools and techniques presented in Chapters 3 to 5 have successfully allowed the creation of touch sensors on various surfaces. However, there remains a challenge in enabling multi-touch sensing on these fabricated sensors without having prior electronic knowledge. To address this challenge, Chapter 6 introduced a touch sensing technique, *Multi-Touch Kit*, which enables even electronics novices to rapidly prototype customized capacitive multi-touch sensors. Unlike existing approaches, the proposed technique utilizes a commodity microcontroller and open-source software, eliminating the need for any specialized hardware.

Fabrication Support for Rich Materials and Complex Geometries

This thesis endeavors to address the challenge of extending fabrication capabilities for the creation of interactive surfaces on various rich materials (Table 7.1). In Chapter 3, a novel sketching technique, *BodyStylus*, was introduced for the interactive fabrication of circuits and sensors on complex geometries, including the human body. This technique allows for physical sketching on soft and deformable substrates, including temporary tattoo sheets and office paper. *BodyStylus* handheld device is equipped to support a range of conductive inks, including gold ink [177], silver pen [195], and regular inks [87].

In Chapter 4, a novel technique, *Print-A-Sketch*, was introduced to facilitate fabrication on diverse surfaces. This approach provides a unique solution to fabricating functional interfaces on surfaces with different textures and absorption properties, including paper, cardboard, textile, plywood, and ceramics, by precisely controlling the ink dispensing process. *Print-A-Sketch* also allows for the creation of functional interfaces on soft, flexible, and stretchable materials, such as kinesiology tape. The *Print-A-Sketch* handheld tool supports the fabrication of both conductive and regular ink, enabling its use in a wide range of applications. Moreover, Chapter 5 presented *RoboSketch*, a fabrication technique for high-resolution printing on a wide range of objects, including soft and deformable surfaces. The *RoboSketch* artifact supports printing with different inks, including conductive, multicolor, UV inks, and black pigment.

Finally, *Multi-Touch Kit*, Chapter 6, enabled multi-touch sensing on customized surfaces and objects, including flat and complex geometries. *Multi-Touch Kit* is capable of supporting sensors fabricated through various technologies such as inkjet printing, hand drawing, hand-stitching, or copper tape.

7.2 Directions for Future Work

In this final section, we will provide an outlook on potential directions for future research. Specifically, we will discuss the unaddressed problems that were identified during the work of this thesis, and propose potential directions for future research to address these challenges.

	Fabrication Tools and Techniques		
	BodyStylus (Chapter 3)	Print-A-Sketch (Chapter 4)	RoboSketch (Chapter 5)
Resolution of Sketches	Low	High	High
Supported Geometries	Complex	Flat	Flat
Surface Material	Tattoo-Sheet, Paper	Wide Range	Wide range
Surface Structure	Soft, Deformable	Soft, Deformable, Stretchable	Soft, Deformable
Supported Ink	Gold, Silver Pen, Regular Stylus	Silver Nanoparticle, Regular	Conductive, Multicolor, Black Pigment, UV

Table 7.1 Overview of fabrication support of contributed physical sketching tools and techniques.

Alternative Fabrication Tools

In this thesis, the emphasis has been on physical sketching using a stylus or printer as the primary fabrication tool. However, the proposed techniques are not limited to these specific tools and can be extended to other fabrication tools. The potential for replacing the stylus and printer with alternative fabrication tools or constructing modular tools presents numerous challenges and opportunities for future exploration. By incorporating alternative tools such as a marker, cutter, miniature laser engraver, or miniature iron for sintering conductive traces into the existing tools of *BodyStylus*, *Print-A-Sketch*, and *RoboSketch*, a wider range of fabrication tasks can be achieved. Additionally, this opens up the possibility of developing autonomous fabrication devices that can work together to perform a specific task.

Physical Sketching on Complex Geometries

The techniques discussed in this thesis introduce novel approaches to fabricating electronic circuits through physical sketching on various surfaces. For instance, *BodyStylus* enables the creation of electronic devices on complex geometries such as the human body through the use of a substrate imprinted with dot patterns, which facilitates the smart pen's position tracking. *Print-A-Sketch* and *RoboSketch* techniques, on the other hand, employ an optical motion sensor for position tracking, thereby enabling direct printing on objects, albeit with the limitation of restricting the sketching environment to planar surfaces. Future research directions could encompass the examination of alternative position-tracking techniques that offer the capability of absolute positioning on a broader spectrum of surface geometries.

Layering Techniques for Customized Interfaces

The focus of the fabrication techniques presented in this thesis is on the creation of single-layer interfaces. This limitation is mainly due to the limitations of the available dielectric ink, which does not have the capacity to isolate the first layer that is printed or support the printing of a second conductive layer. Nevertheless, with the progress in material and ink technology, it is conceivable to envision the creation of multi-layer interfaces with improved capabilities in the future. The integration of a variety of sensors and actuators that require multi-layering, including multi-touch sensors, electro-tactile output, and electroluminescence displays, offers the possibility of delivering a more enriching and engaging user experience. Hence, the examination of multi-layering presents a promising direction for future research and holds the potential to lead to significant advancements in the field of customized interfaces.

Expanding Capabilities in Multi-Touch Sensing Technique

This thesis presents a touch-sensing technique that has the capability to support sensors with up to 16 x 16 electrodes. Incresing the number of electrodes provides an opportunity for a wider coverage area and the possibility of new applications. The implementation of *Multi-Touch Kit* employs the classical two-layered diamond pattern, which is a widely-used approach in mutual-capacitive touch sensing [42; 127]. The dimensions of the electrodes range from 4 x 4 mm to 6 x 6 mm, making it suitable for high-resolution finger touch sensing. However, larger electrode dimensions exceeding 6 x 6 mm could enable the detection of larger body parts such as hands or bare feet, thereby opening up new opportunities for the creation of multi-touch sensor-covered walls and floors [260].

In addition, the touch-sensing technique described in Chapter 6 was implemented using the Atmel megaAVR family of microcontrollers. Future work includes the expansion of hardware support for the open-source library by incorporating other commonly used platforms, such as Teensy and Raspberry PI.

This thesis has presented several contributions toward physical sketching tools and techniques for customized sensate surfaces. It is hoped that the insights, tools, and techniques presented in this thesis will serve as a valuable resource for researchers, makers, and designers working in this field.

List of Figures

References

[1] Code of federal regulations, title 47: Telecommunication, 15.23 home-built devices. http://www.ecfr.gov/, 2019. Retrieved: 04-04-2019.

[2] LED Henna Tattoos Make You Look Like a Bohemian Cyborg. https://www.vice.com/en_uk/article/53w93x/led-henna-tattoos-make-you-look-like-a-bohemian-cyborg, 2020. Accessed: 2020-04-01.

[3] Adafruit. 2-axis joystick, 2022. Accessed: 2022-06-02.

[4] Linear Micro Servo For Ultra-Micro aircraft. https://www.amazon.com/Digital-Loading-Actuator-Ultra-Micro-Aircraft/dp/B07G34BT9N, 2020. Accessed: 2020-03-16.

[5] Robert Aish. Designscript: Origins, explanation, illustration. In Christoph Gengnagel, Axel Kilian, Norbert Palz, and Fabian Scheurer, editors, *Computational Design Modelling*, pages 1–8, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[6] H. Akhtar and R. Kakarala. A methodology for evaluating accuracy of capacitive touch sensing grid patterns. *Journal of Display Technology*, 10(8):672–682, Aug 2014.

[7] Kristina Andersen, Ron Wakkary, Laura Devendorf, and Alex McLean. Digital crafts-machine-ship: Creative collaborations with machines. *Interactions*, 27(1):30–35, dec 2019.

[8] AutoTrace. Converts bitmap to vector graphics, 2022. Accessed: 2022-06-02.

[9] J. Bae, H. Cho, K. Song, H. Lee, and H. Yoo. The signal transmission mechanism on the surface of human body for body channel communication. *IEEE Transactions on Microwave Theory and Techniques*, 60(3):582–593, March 2012.

[10] Frank Beck and Bent Stumpe. Two devices for operator interaction in the central control of the new cern accelerator. Technical report, CERN, 1973.

[11] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3479–3487, Washington, DC, USA, 2015. IEEE Computer Society.

[12] Joanna Bergstrom-Lehtovirta, Sebastian Boring, and Kasper Hornbæk. Placing and recalling virtual items on the skin. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 1497–1507, New York, NY, USA, 2017. ACM.

[13] Tracey Booth, Simone Stumpf, Jon Bird, and Sara Jones. Crossed wires: Investigating the problems of end-user developers in a physical computing task. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, page 3485–3497, New York, NY, USA, 2016. Association for Computing Machinery.

[14] BOTSY. Botsy- the wall drawing robot, 2007. Accessed: 2022-06-02.

[15] Audrey Briot, Cedric Honnet, and Paul Strohmeier. Stymphalian birds - exploring the aesthetics of a hybrid textile. In *Companion Publication of the 2020 ACM Designing Interactive Systems Conference*, DIS' 20 Companion, page 437–440, New York, NY, USA, 2020. Association for Computing Machinery.

[16] Marion Buchenau and Jane Fulton Suri. Experience prototyping. In *Proceedings of the 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, DIS '00, page 424–433, New York, NY, USA, 2000. Association for Computing Machinery.

[17] Leah Buechley, Sue Hendrix, and Mike Eisenberg. Paints, paper, and programs: First steps toward the computational sketchbook. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, TEI '09, page 9–12, New York, NY, USA, 2009. Association for Computing Machinery.

[18] Leah Buechley and Hannah Perner-Wilson. Crafting technology: Reimagining the processes, materials, and cultures of electronics. *ACM Trans. Comput.-Hum. Interact.*, 19(3), October 2012.

[19] BYK. Substrate wetting, 2020. Accessed: 2021-02-02.

[20] Varun Perumal C and Daniel Wigdor. Printem: Instant printed circuit boards with standard office printers and amp; inks. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and amp; Technology*, UIST '15, page 243–251, New York, NY, USA, 2015. Association for Computing Machinery.

[21] Bradley Camburn, Vimal Viswanathan, Julie Linsey, David Anderson, Daniel Jensen, Richard Crawford, Kevin Otto, and Kristin Wood. Design prototyping methods: state of the art in strategies, techniques, and guidelines. *Design Science*, 3:e13, 2017.

[22] Daniel Campos Zamora, Liang He, Yueqian Zhang, Xuhai Xu, Jennifer Mankoff, and Jon E. Froehlich. Sprintr: Towards in-situ personal fabrication using a mobile 3d printer. In *Proceedings of the 7th Annual ACM Symposium on Computational Fabrication*, SCF '22, New York, NY, USA, 2022. Association for Computing Machinery.

[23] Vanessa Julia Carpenter, Amanda Willis, Nikolaj "Dzl" Møbius, and Dan Overholt. Electronic kintsugi. In Kohei Arai, Rahul Bhatia, and Supriya Kapoor, editors, *Proceedings of the Future Technologies Conference (FTC) 2018*, pages 104–121, Cham, 2019. Springer International Publishing.

[24] Sarah Chadwick and Mamta Shah. Tattoos: ancient body art may assist in medical emergencies. *European Journal of Pediatrics*, 172(7):995–995, 2013.

[25] Tingyu Cheng, Bu Li, Yang Zhang, Yunzhi Li, Charles Ramey, Eui Min Jung, Yepu Cui, Sai Ganesh Swaminathan, Youngwook Do, Manos Tentzeris, Gregory D. Abowd, and HyunJoo Oh. Duco: Autonomous large-scale direct-circuit-writing (dcw) on vertical everyday surfaces using a scalable hanging plotter. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 5(3), sep 2021.

[26] Tingyu Cheng, Koya Narumi, Youngwook Do, Yang Zhang, Tung D. Ta, Takuya Sasatani, Eric Markvicka, Yoshihiro Kawahara, Lining Yao, Gregory D. Abowd, and HyunJoo Oh. Silver Tape: Inkjet-Printed Circuits Peeled-and-Transferred on Versatile Substrates. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(1):1–17, March 2020.

[27] chibitronics. Expressive technology with paper circuits, 2021. Accessed: 2021-02-02.

[28] Youngkyung Choi, Neung Ryu, Myung Jin Kim, Artem Dementyev, and Andrea Bianchi. Bodyprinter: Fabricating circuits directly on the skin at arbitrary locations using a wearable compact plotter. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, UIST '20, page 554–564, New York, NY, USA, 2020. Association for Computing Machinery.

[29] Colop. https://emark.colop.com/, 2021. Accessed: 2021-06-15.

[30] Bare Conductive. https://www.bareconductive.com, 2020. Accessed: 2020-02-15.

[31] Nigel Cross. *Designerly ways of knowing*. Springer, 2006.

[32] Gerard Cummins and Marc P.Y. Desmulliez. Inkjet printing of conductive materials: a review. *Circuit World*, 38(4):193–213, January 2012. Publisher: Emerald Group Publishing Limited.

[33] Nadine Dafrawy. 5 complicated geometric henna designs completely simplified (a step-by-step guide), 2019. Accessed: 2020-09-01.

[34] Ella Dagan, Elena Márquez Segura, Ferran Altarriba Bertran, Miguel Flores, Robb Mitchell, and Katherine Isbister. Design framework for social wearables. In *Proceedings of the 2019 on Designing Interactive Systems Conference*, DIS '19, page 1001–1015, New York, NY, USA, 2019. Association for Computing Machinery.

[35] Nicholas Davis, Chih-PIn Hsiao, Kunwar Yashraj Singh, Lisa Li, and Brian Magerko. Empirically studying participatory sense-making in abstract drawing with a co-creative cognitive agent. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, IUI '16, page 196–207, New York, NY, USA, 2016. Association for Computing Machinery.

[36] Burke Davison. Techniques for robust touch sensing design. *AN1334 Microchip Technology Inc*, page 53, 2010.

[37] Joseph DeGol, Mani Golparvar-Fard, and Derek Hoiem. Geometry-informed material recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1554–1562, Washington, DC, USA, 2016. IEEE Computer Society.

[38] Scribit Design. Scribit, 2019. Accessed: 2022-06-02.

[39] Sebastian Deterding, Jonathan Hook, Rebecca Fiebrink, Marco Gillies, Jeremy Gow, Memo Akten, Gillian Smith, Antonios Liapis, and Kate Compton. Mixed-initiative creative interfaces. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, page 628–635, New York, NY, USA, 2017. Association for Computing Machinery.

[40] Olaf Diegel, Aparna Badve, Glen Bright, Johan Potgieter, and Sylvester Tlale. Improved mecanum wheel design for omni-directional robots. In *Proc. 2002 Australasian conference on robotics and automation, Auckland*, pages 117–121, 2002.

[41] Georg Freitag Markus Wacker Dietrich Kammer, Mandy Keck. Taxonomy and overview of multi-touch frameworks: Architecture, scope and features. In *Workshop on Engineering Patterns for Multitouch Interfaces*, 2010.

[42] Paul Dietz and Darren Leigh. Diamondtouch: A multi-user touch technology. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, UIST '01, pages 219–226, New York, NY, USA, 2001. ACM.

[43] Leopold Dorfer, Maximilian Moser, Frank Bahr, Konrad Spindler, Eduard Egarter-Vigl, Sonia Giullén, Gottfried Dohr, and Thomas Kenner. A medical report from the stone age? *The Lancet*, 354(9183):1023–1025, 1999.

[44] Miroslav Dulik and Libor Ladanyi. Surface detection and recognition using infrared light. In *2014 ELEKTRO*, volume 1, pages 159–164, Washington, DC, USA, 2014. IEEE Computer Society.

[45] EASYSMX. Esm-9013 wireless controller, 2022. Accessed: 2022-06-02.

[46] Simon Eccles. The ultimate guide for printhead technologies, 2017. Accessed: 2021-02-02.

[47] C.W. Eldridge. The tattoo archive – stencils, 2004.

[48] Epson. Photo paper glossy, 2022. Accessed: 2022-06-02.

[49] Katherine M Everitt, Meredith Ringel Morris, AJ Bernheim Brush, and Andrew D Wilson. Docudesk: An interactive surface for creating and rehydrating many-to-many linkages among paper and digital documents. In *2008 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems*, pages 25–28, New York City, 2008. IEEE, IEEE.

[50] Nicholas Fellion, Thomas Pietrzak, and Audrey Girouard. Flexstylus: Leveraging bend input for pen interaction. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, page 375–385, New York, NY, USA, 2017. Association for Computing Machinery.

[51] Kenneth D. Forbus, Ronald W. Ferguson, and Jeffery M. Usher. Towards a computational model of sketching. In *Proceedings of the 6th International Conference on Intelligent User Interfaces*, IUI '01, page 77–83, New York, NY, USA, 2001. Association for Computing Machinery.

[52] Frikk Fossdal, Rogardt Heldal, and Nadya Peek. Interactive digital fabrication machine control directly within a cad environment. In *Proceedings of the 6th Annual ACM Symposium on Computational Fabrication*, SCF '21, New York, NY, USA, 2021. Association for Computing Machinery.

[53] Rachel Freire, Cedric Honnet, and Paul Strohmeier. Second skin: An exploration of etextile stretch circuits on the body. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '17, page 653–658, New York, NY, USA, 2017. Association for Computing Machinery.

[54] L Furey. Archaeological evidence for tattooing in polynesia and micronesia. In Lars Krutak and Aaron Deter-Wolf, editors, *Ancient Ink: The archaeology of tattooing*, chapter 11, pages 159–184. University of Washington Press, Washington, 2017.

[55] Madeline Gannon, Tovi Grossman, and George Fitzmaurice. Tactum: A skin-centric approach to digital design and fabrication. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 1779–1788, New York, NY, USA, 2015. ACM.

[56] Madeline Gannon, Tovi Grossman, and George Fitzmaurice. Exoskin: On-body fabrication. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 5996–6007, New York, NY, USA, 2016. ACM.

[57] Impasto gel. https://lascaux.ch/en/products/painting-mediums/lascaux-impasto-gels, 2020. Accessed: 2020-09-06.

[58] Nan-Wei Gong, Nan Zhao, and Joseph A. Paradiso. A customizable sensate surface for music control. In *New Interfaces for Musical Expression*, 2012.

[59] Tony Gray. *Projected Capacitive Touch*. Springer International Publishing, Cham, 2019.

[60] Daniel Groeger and Jürgen Steimle. ObjectSkin: Augmenting Everyday Objects with Hydroprinted Touch Sensors and Displays. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(4):1–23, January 2018.

[61] T. Grosse-Puppendahl, Y. Berghoefer, A. Braun, R. Wimmer, and A. Kuijper. Opencapsense: A rapid prototyping toolkit for pervasive interaction using capacitive sensing. In *2013 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 152–159, March 2013.

[62] Tobias Grosse-Puppendahl, Andreas Braun, and Xavier Dellangnol. Prototyping capacitive sensing applications with opencapsense. *GetMobile: Mobile Comp. and Comm.*, 20(2):16–21, October 2016.

[63] Tobias Grosse-Puppendahl, Christian Holz, Gabe Cohn, Raphael Wimmer, Oskar Bechtold, Steve Hodges, Matthew S. Reynolds, and Joshua R. Smith. Finding common ground: A survey of capacitive sensing in human-computer interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '17, New York, NY, USA, 2017. ACM.

[64] Sean G. Gustafson, Bernhard Rabe, and Patrick M. Baudisch. Understanding palm-based imaginary interfaces: The role of visual and tactile cues when browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 889–898, New York, NY, USA, 2013. Association for Computing Machinery.

[65] David Ha and Douglas Eck. A neural representation of sketch drawings, 2017.

[66] Nur Al-huda Hamdan, Simon Voelker, and Jan Borchers. Sketch&stitch: Interactive embroidery for e-textiles. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, New York, NY, USA, 2018. Association for Computing Machinery.

[67] Nur Al-huda Hamdan, Adrian Wagner, Simon Voelker, Jürgen Steimle, and Jan Borchers. Springlets: Expressive, flexible and silent on-skin tactile interfaces. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pages 488:1–488:14, New York, NY, USA, 2019. ACM.

[68] Jefferson Y. Han. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, UIST '05, pages 115–118, New York, NY, USA, 2005. ACM.

[69] Teng Han, Shubhi Bansal, Xiaochen Shi, Yanjun Chen, Baogang Quan, Feng Tian, Hongan Wang, and Sriram Subramanian. Hapbead: On-skin microfluidic haptic interface using tunable bead. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–10, New York, NY, USA, 2020. Association for Computing Machinery.

[70] Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. Omnitouch: Wearable multi-touch interaction everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 441–450, New York, NY, USA, 2011. ACM.

[71] Chris Harrison and Haakon Faste. Implications of location and touch for on-body projected interfaces. In *Proceedings of the 2014 Conference on Designing Interactive Systems*, DIS '14, pages 543–552, New York, NY, USA, 2014. Association for Computing Machinery.

[72] Chris Harrison, Julia Schwarz, and Scott E. Hudson. Tapsense: enhancing finger interaction on touch surfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST '11*, page 627, New York, New York, USA, 2011. ACM Press.

[73] Chris Harrison, Desney Tan, and Dan Morris. Skinput: Appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 453–462, New York, NY, USA, 2010. ACM.

[74] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[75] Steve Hodges, Nicolas Villar, Nicholas Chen, Tushar Chugh, Jie Qi, Diana Nowacka, and Yoshihiro Kawahara. Circuit stickers: Peel-and-stick construction of interactive electronic prototypes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 1743–1746, New York, NY, USA, 2014. Association for Computing Machinery.

[76] David Holman, Nicholas Fellion, and Roel Vertegaal. Sensing touch using resistive graphs. In *Proceedings of the 2014 Conference on Designing Interactive Systems*, DIS '14, pages 195–198, New York, NY, USA, 2014. ACM.

[77] Lars Erik Holmquist. Sketching in hardware. *Interactions*, 13(1):47–60, January 2006.

[78] Christian Holz and Patrick Baudisch. Understanding touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 2501–2510, New York, NY, USA, 2011. ACM.

[79] Cedric Honnet, Hannah Perner-Wilson, Marc Teyssier, Bruno Fruchard, Jürgen Steimle, Ana C. Baptista, and Paul Strohmeier. *PolySense: Augmenting Textiles with Electrical Functionality Using In-Situ Polymerization*, page 1–13. Association for Computing Machinery, New York, NY, USA, 2020.

[80] Eric Horvitz. Uncertainty, action, and interaction: In pursuit of mixed-initiative computing. *IEEE Intelligent Systems*, 14(5):17–20, 1999.

[81] HowToMechatronics.com. Arduino mecanum wheels robot, 2022. Accessed: 2022-06-02.

[82] Takeo Igarashi, Satoshi Matsuoka, Sachiko Kawachiya, and Hidehiko Tanaka. Interactive beautification: A technique for rapid geometric design. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, UIST '97, page 105–114, New York, NY, USA, 1997. Association for Computing Machinery.

[83] PHIDGETS INC. phidgets, 2016. Accessed: 2021-02-02.

[84] Freehand Ink. https://inkbox.com/freehand-ink, 2020. Accessed: 2020-04-01.

[85] Vishay Intertechnology. Graphic oled, 2017. Accessed: 2021-06-02.

[86] iRobot. Root, 2022. Accessed: 2022-06-02.

[87] Rotring Isograph. https://www.rotring.com/de/technische-zeichengerate/50-isograph-4006856151008.html, 2020. Accessed: 2020-02-15.

[88] Jennifer Jacobs, Joel Brandt, Radomír Mech, and Mitchel Resnick. Extending manual drawing practices with artist-centric programming tools. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–13, New York, NY, USA, 2018. Association for Computing Machinery.

[89] Jennifer Jacobs, Mitchel Resnick, and Leah Buechley. Dresscode: supporting youth in computational design and making. In *Proceedings of Constructionism 2014 Conference. Vienna, Austria*, volume 10, 2014.

[90] Brett Jones, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, Nikunj Raghuvanshi, and Lior Shapira. Roomalive: Magical experiences enabled by scalable, adaptive projector-camera units. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 637–644, New York, NY, USA, 2014. ACM.

[91] Lee Jones. The e-darning sampler: Exploring e-textile repair with darning looms. In *Proceedings of the Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '21, New York, NY, USA, 2021. Association for Computing Machinery.

[92] Kalatmak. Election special mehndi design / indian election 2019 henna art, mar 2019.

[93] Martin Kaltenbrunner and Ross Bencina. reactivision: A computer-vision framework for table-based tangible interaction. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, TEI '07, pages 69–74, New York, NY, USA, 2007. ACM.

[94] Hsin-Liu (Cindy) Kao, Deborah Ajilo, Oksana Anilionyte, Artem Dementyev, Inrak Choi, Sean Follmer, and Chris Schmandt. Exploring interactions and perceptions of kinetic wearables. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, DIS '17, page 391–396, New York, NY, USA, 2017. Association for Computing Machinery.

[95] Hsin-Liu (Cindy) Kao, Miren Bamforth, David Kim, and Chris Schmandt. Skinmorph: Texture-tunable on-skin interface through thin, programmable gel. In *Proceedings of the 2018 ACM International Symposium on Wearable Computers*, ISWC '18, pages 196–203, New York, NY, USA, 2018. ACM.

[96] Hsin-Liu (Cindy) Kao, Christian Holz, Asta Roseway, Andres Calvo, and Chris Schmandt. Duoskin: Rapidly prototyping on-skin user interfaces using skin-friendly materials. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*, ISWC '16, pages 16–23, New York, NY, USA, 2016. ACM.

[97] Hsin-Liu (Cindy) Kao, Manisha Mohan, Chris Schmandt, Joseph A. Paradiso, and Katia Vega. Chromoskin: Towards interactive cosmetics using thermochromic pigments. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '16, pages 3703–3706, New York, NY, USA, 2016. ACM.

[98] Yoshihiro Kawahara, Steve Hodges, Benjamin S. Cook, Cheng Zhang, and Gregory D. Abowd. Instant inkjet circuits: Lab-based inkjet printing to support rapid prototyping of ubicomp devices. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, page 363–372, New York, NY, USA, 2013. Association for Computing Machinery.

[99] Arshad Khan, Joan Sol Roo, Tobias Kraus, and Jürgen Steimle. Soft inkjet circuits: Rapid multi-material fabrication of soft circuits using a commodity inkjet printer. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, pages 341–354, New York, NY, USA, 2019. Association for Computing Machinery.

[100] Soheil Kianzad, Yuxiang Huang, Robert Xiao, and Karon E. MacLean. Phasking on paper: Accessing a continuum of physically assisted sketching. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–12, New York, NY, USA, 2020. Association for Computing Machinery.

[101] Jeeeun Kim, Haruki Takahashi, Homei Miyashita, Michelle Annett, and Tom Yeh. Machines as co-designers: A fiction on the future of human-fabrication machine interaction. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, page 790–805, New York, NY, USA, 2017. Association for Computing Machinery.

[102] Jeeeun Kim, Clement Zheng, Haruki Takahashi, Mark D Gross, Daniel Ashbrook, and Tom Yeh. Compositional 3d printing: Expanding & supporting workflows towards continuous fabrication. In *Proceedings of the 2nd Annual ACM Symposium on Computational Fabrication*, SCF '18, New York, NY, USA, 2018. Association for Computing Machinery.

[103] Jeonghyun Kim, Anthony Banks, Huanyu Cheng, Zhaoqian Xie, Sheng Xu, Kyung-In Jang, Jung Woo Lee, Zhuangjian Liu, Philipp Gutruf, Xian Huang, Pinghung Wei, Fei Liu, Kan Li, Mitul Dalal, Roozbeh Ghaffari, Xue Feng, Yonggang Huang, Sanjay Gupta, Ungyu Paik, and John A. Rogers. Epidermal electronics with advanced capabilities in near-field communication. *Small*, 11(8):906–912, 2015.

[104] Konstantin Klamka and Raimund Dachselt. Illumipaper: Illuminated interactive paper. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 5605–5618, New York, NY, USA, 2017. Association for Computing Machinery.

[105] Konstantin Klamka, Raimund Dachselt, and Jürgen Steimle. Rapid iron-on user interfaces: Hands-on fabrication of interactive textile prototypes. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–14, New York, NY, USA, 2020. Association for Computing Machinery.

[106] Marion Koelle, Swamy Ananthanarayan, and Susanne Boll. Social acceptability in hci: A survey of methods, measures, and design strategies. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–19, New York, NY, USA, 2020. Association for Computing Machinery.

[107] Lars F. Krutak. *Spiritual Skin: Magical Tattoos and Scarification: Wisdom, Healing, Shamanic Power, Protection.* Ed. Reuss, Glattbach, 2012.

[108] Stacey Kuznetsov, Piyum Fernando, Emily Ritter, Cassandra Barrett, Jennifer Weiler, and Marissa Rohr. Screenprinting and tei: Supporting engagement with steam

through diy fabrication of smart materials. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '18, page 211–220, New York, NY, USA, 2018. Association for Computing Machinery.

[109] Emily A. Lai, Eric A. Nesrsta, Jason M. Bodiford, and Ranu R. Jain. Medical alert tattoos-a pinch of style along with patient safety. *Pediatric Anesthesia*, 28(12):1154–1155, 2018.

[110] James A. Landay. Silk: Sketching interfaces like krazy. In *Conference Companion on Human Factors in Computing Systems*, CHI '96, page 398–399, New York, NY, USA, 1996. Association for Computing Machinery.

[111] Thomas Langerak, Juan José Zárate, Velko Vechev, David Lindlbauer, Daniele Panozzo, and Otmar Hilliges. *Optimal Control for Electromagnetic Haptic Guidance Systems*, page 951–965. Association for Computing Machinery, New York, NY, USA, 2020.

[112] Thomas Langerak, Juan Jose Zarate, Velko Vechev, Daniele Panozzo, and Otmar Hilliges. Dynamic drawing guidance via electromagnetic haptic feedback. *CoRR*, abs/1906.11753:207–217, 2019.

[113] Johnny C. Lee, Paul H. Dietz, Darren Leigh, William S. Yerazunis, and Scott E. Hudson. Haptic pen: A tactile feedback stylus for touch screens. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, page 291–294, New York, NY, USA, 2004. Association for Computing Machinery.

[114] Kang-Hee Lee and Jong-Hwan Kim. Multi-robot cooperation-based mobile printer system. *Robotics and Autonomous Systems*, 54(3):193–204, 2006.

[115] Yong Jae Lee, C. Lawrence Zitnick, and Michael F. Cohen. Shadowdraw: Real-time user guidance for freehand drawing. *ACM Trans. Graph.*, 30(4), jul 2011.

[116] Darren Leigh, Clifton Forlines, Ricardo Jota, Steven Sanders, and Daniel Wigdor. High rate, low-latency multi-touch sensing with simultaneous orthogonal multiplexing. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 355–364, New York, NY, USA, 2014. ACM.

[117] Jingyi Li, Joel Brandt, Radomír Mech, Maneesh Agrawala, and Jennifer Jacobs. Supporting visual artists in programming through direct inspection and control of program execution. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–12, New York, NY, USA, 2020. Association for Computing Machinery.

[118] Jingyi Li, Jennifer Jacobs, Michelle Chang, and Björn Hartmann. Direct and immediate drawing with cnc machines. In *Proceedings of the 1st Annual ACM Symposium on Computational Fabrication*, SCF '17, New York, NY, USA, 2017. Association for Computing Machinery.

[119] Yuyu Lin, Jiahao Guo, Yang Chen, Cheng Yao, and Fangtian Ying. *It Is Your Turn: Collaborative Ideation With a Co-Creative Robot through Sketch*, page 1–14. Association for Computing Machinery, New York, NY, USA, 2020.

[120] lingib. Omni wheel cnc plotter v2, 2019. Accessed: 2022-11-22.

[121] Joanne Lo, Doris Jung Lin Lee, Nathan Wong, David Bui, and Eric Paulos. Skintillates: Designing and creating epidermal interactions. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, DIS '16, pages 853–864, New York, NY, USA, 2016. ACM.

[122] Joanne Lo, Doris Jung Lin Lee, Nathan Wong, David Bui, and Eric Paulos. Skintillates: Designing and creating epidermal interactions. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, DIS '16, pages 853–864, New York, NY, USA, 2016. ACM.

[123] Joanne Lo, Cesar Torres, Isabel Yang, Jasper O'Leary, Danny Kaufman, Wilmot Li, Mira Dontcheva, and Eric Paulos. Aesthetic electronics: Designing, sketching, and fabricating circuits through digital exploration. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 665–676, New York, NY, USA, 2016. Association for Computing Machinery.

[124] Eric Markvicka, Guanyun Wang, Yi-Chin Lee, Gierad Laput, Carmel Majidi, and Lining Yao. Electrodermis: Fully untethered, stretchable, and highly-customizable electronic bandages. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pages 632:1–632:10, New York, NY, USA, 2019. ACM.

[125] David A. Mellis, Sam Jacoby, Leah Buechley, Hannah Perner-Wilson, and Jie Qi. Microcontrollers as material: Crafting circuits with paper, conductive ink, electronic components, and an "untoolkit". In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction*, TEI '13, pages 83–90, New York, NY, USA, 2013. Association for Computing Machinery.

[126] James J. Fitzgibbon Michael C. Brenner. Surface acoustic wave touch panel system, 22 1985. US Patent US4644100A.

[127] Microchip. Sensor design guidelines, January 2012. Retrieved April 4, 2019 from http://ww1.microchip.com/downloads/en/DeviceDoc/FAQs%20-%20Sensor%20Design%20Guidelines.pdf.

[128] Mitsubishi. Mitsubishi nanobenefit 3g series, 2020. Accessed: 2021-02-02.

[129] Daniela Mitterberger, Selen Ercan Jenny, Lauren Vasey, Ena Lloret-Fritschi, Petrus Aejmelaeus-Lindström, Fabio Gramazio, and Matthias Kohler. Interactive robotic plastering: Augmented interactive design and fabrication for on-site robotic plastering. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery.

[130] Wireless Charging Module. https://www.seeedstudio.com/Wireless-Charging-Module-5V-1A-p-1912.html, 2020. Accessed: 2020-09-01.

[131] Stefanie Mueller, Pedro Lopes, and Patrick Baudisch. Interactive construction: Interactive fabrication of functional mechanical devices. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 599–606, New York, NY, USA, 2012. ACM.

[132] Stefanie Mueller, Anna Seufert, Huaishu Peng, Robert Kovacs, Kevin Reuss, François Guimbretière, and Patrick Baudisch. Formfab: Continuous interactive fabrication. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '19, pages 315–323, New York, NY, USA, 2019. Association for Computing Machinery.

[133] Sachith Muthukumarana, Don Samitha Elvitigala, Juan Pablo Forero Cortes, Denys J.C. Matthies, and Suranga Nanayakkara. Touch me gently: Recreating the perception of touch using a shape-memory alloy matrix. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–12, New York, NY, USA, 2020. Association for Computing Machinery.

[134] Ken Nakagaki and Yasuaki Kakehi. Comp*pass: A compass-based drawing interface. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, page 447–450, New York, NY, USA, 2014. Association for Computing Machinery.

[135] Nanotec. Linear actuators with stepper motors, 2021. Accessed: 2021-06-25.

[136] Koya Narumi, Steve Hodges, and Yoshihiro Kawahara. Conductar: An augmented reality based tool for iterative design of conductive ink circuits. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, page 791–800, New York, NY, USA, 2015. Association for Computing Machinery.

[137] Koya Narumi, Steve Hodges, and Yoshihiro Kawahara. Conductar: An augmented reality based tool for iterative design of conductive ink circuits. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, pages 791–800, New York, NY, USA, 2015. Association for Computing Machinery.

[138] Koya Narumi, Xinyang Shi, Steve Hodges, Yoshihiro Kawahara, Shinya Shimizu, and Tohru Asami. Circuit eraser: A tool for iterative design with conductive ink. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, page 2307–2312, New York, NY, USA, 2015. Association for Computing Machinery.

[139] Aditya Shekhar Nittala, Arshad Khan, Klaus Kruttwig, Tobias Kraus, and Jürgen Steimle. Physioskin: Rapid fabrication of skin-conformal physiological interfaces. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–10, New York, NY, USA, 2020. Association for Computing Machinery.

[140] Aditya Shekhar Nittala, Anusha Withana, Narjes Pourjafarian, and Jürgen Steimle. Multi-touch skin: A thin and flexible multi-touch sensor for on-skin input. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 33:1–33:12, New York, NY, USA, 2018. ACM.

[141] Sandy Noble. Polargraph, 2019. Accessed: 2022-11-22.

[142] NovaCentrix. Metalon conductive inks for printed electronics, 2018. Accessed: 2021-02-02.

[143] Changhoon Oh, Jungwoo Song, Jinhan Choi, Seonghyeon Kim, Sungwoo Lee, and Bongwon Suh. I lead, you help but only with enough details: Understanding user experience of co-creation with artificial intelligence. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–13, New York, NY, USA, 2018. Association for Computing Machinery.

[144] Hyunjoo Oh, Tung D. Ta, Ryo Suzuki, Mark D. Gross, Yoshihiro Kawahara, and Lining Yao. Pep (3d printed electronic papercrafts): An integrated approach for 3d sculpting paper-based electronic devices. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–12, New York, NY, USA, 2018. Association for Computing Machinery.

[145] Simon Olberding, Sergio Soto Ortega, Klaus Hildebrandt, and Jürgen Steimle. Foldio: Digital fabrication of interactive and shape-changing objects with foldable printed electronics. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, UIST '15, pages 223–232, New York, NY, USA, 2015. Association for Computing Machinery.

[146] Simon Olberding and Jürgen Steimle. Towards understanding erasing-based interactions: Adding erasing capabilities to anoto pens. In *1st International Workshop on Paper Computing*, pages 20–23, Copenhagen, Denmark, 2010. PaperComp 2010.

[147] Simon Olberding, Jürgen Steimle, Suranga Nanayakkara, and Pattie Maes. Clouddrops: Stamp-sized pervasive displays for situated awareness of web-based information. In *Proceedings of the 4th International Symposium on Pervasive Displays*, PerDis '15, pages 47–53, New York, NY, USA, 2015. ACM.

[148] Simon Olberding, Michael Wessely, and Jürgen Steimle. PrintScreen: fabricating highly customizable thin-film touch-displays. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*, pages 281–290, Honolulu, Hawaii, USA, 2014. ACM Press.

[149] M. Packard and E. Kwei. *Henna Sourcebook: Over 1,000 traditional designs and modern interpretations for body decorating.* Race Point Publishing, New York, 2012.

[150] Sunnyscopa Temporary Laser Tattoo Paper. https://www.amazon.de/-/en/gp/product/B07FY81ZLG/ref=ppx_yo_dt_b_asin_title_o09_s00?ie=UTF8&psc=1, 2020. Accessed: 2020-09-01.

[151] Joseph A. Paradiso and Neil Gershenfeld. Musical Applications of Electric Field Sensing. *Computer Music Journal*, 21(2):69, 1997.

[152] Ncode PDF. https://www.neosmartpen.com/en/ncode-pdf/, 2020. Accessed: 2020-09-01.

[153] Hao-Lun Peng and Yoshihiro Watanabe. High-speed human arm projection mapping with skin deformation. In *SIGGRAPH Asia 2020 Emerging Technologies*, SA '20, New York, NY, USA, 2020. Association for Computing Machinery.

[154] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. Roma: Interactive fabrication with augmented reality and a robotic 3d printer. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, New York, NY, USA, 2018. Association for Computing Machinery.

[155] Huaishu Peng, Amit Zoran, and François V. Guimbretière. D-coil: A hands-on approach to digital 3d models design. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 1807–1815, New York, NY, USA, 2015. Association for Computing Machinery.

[156] Hannah Perner-Wilson, Leah Buechley, and Mika Satomi. Handcrafting textile interfaces from a kit-of-no-parts. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '11, page 61–68, New York, NY, USA, 2010. Association for Computing Machinery.

[157] PixArt. Pmw3360dm-t2qu: Optical gaming navigation chip, 2016. Accessed: 2021-06-25.

[158] Xaar plc. Adaptable printhead with trouble-free integration, 2020. Accessed: 2021-02-02.

[159] Pololu. 250:1 micro metal gearmotor hp 6v, 2022. Accessed: 2022-06-02.

[160] Pololu. Magnetic encoder pair kit for micro metal gearmotors, 2022. Accessed: 2022-06-02.

[161] Ivan Poupyrev, Nan-Wei Gong, Shiho Fukuhara, Mustafa Emre Karagozler, Carsten Schwesig, and Karen E. Robinson. Project jacquard: Interactive digital textiles at scale. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 4216–4227, New York, NY, USA, 2016. ACM.

[162] Narjes Pourjafarian, Marion Koelle, Bruno Fruchard, Sahar Mavali, Konstantin Klamka, Daniel Groeger, Paul Strohmeier, and Jürgen Steimle. Bodystylus: Freehand on-body design and fabrication of epidermal interfaces. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery.

[163] Narjes Pourjafarian, Marion Koelle, Fjolla Mjaku, Paul Strohmeier, and Jürgen Steimle. Print-a-sketch: A handheld printer for physical sketching of circuits and sensors on everyday surfaces. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery.

[164] Narjes Pourjafarian, Fjolla Mjaku, Marion Koelle, Martin Schmitz, Jan Borchers, and Jürgen Steimle. Handheld tools unleashed: Mixed-initiative physical sketching with a robotic printer. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA, 2023. Association for Computing Machinery.

[165] Narjes Pourjafarian, Anusha Withana, Joseph A. Paradiso, and Jürgen Steimle. Multi-touch kit: A do-it-yourself technique for capacitive multi-touch sensing using a commodity microcontroller. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, page 1071–1083, New York, NY, USA, 2019. Association for Computing Machinery.

[166] Prinker. https://prinker-europe.com/en/, 2020. Accessed: 2020-02-15.

[167] The Tattoo Archive Project. https://artsandculture.google.com/exhibit/history-ink/wQx72HUG, 2020. Accessed: 2020-04-01.

[168] Bridget Punsalang. How to henna a pregnant belly, workshop at the 4th annual henna gathering, 2011. Accessed: 2020-09-01.

[169] Jie Qi and Leah Buechley. Electronic popables: Exploring paper-based computing through an interactive pop-up book. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '10, page 121–128, New York, NY, USA, 2010. Association for Computing Machinery.

[170] Jie Qi and Leah Buechley. Sketching in circuits: Designing and building electronics on paper. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, page 1713–1722, New York, NY, USA, 2014. Association for Computing Machinery.

[171] Jie Qi, Leah Buechley, Andrew "bunnie" Huang, Patricia Ng, Sean Cross, and Joseph A. Paradiso. Chibitronics in the wild: Engaging new communities in creating technology with paper electronics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–11, New York, NY, USA, 2018. Association for Computing Machinery.

[172] Jie Qi, Andrew "bunnie" Huang, and Joseph Paradiso. Crafting technology with circuit stickers. In *Proceedings of the 14th International Conference on Interaction Design and Children*, IDC '15, page 438–441, New York, NY, USA, 2015. Association for Computing Machinery.

[173] Raf Ramakers, Kashyap Todi, and Kris Luyten. Paperpulse: An integrated approach for embedding electronics in paper designs. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 2457–2466, New York, NY, USA, 2015. ACM.

[174] Maya Ranganathan and Bernadette Lobo. Localizing the global: Analysis of nationalist ideologies in mnc advertisements in indian tv. *Nationalism and Ethnic Politics*, 14(1):117–142, 2008.

[175] Majken K. Rasmussen, Giovanni M. Troiano, Marianne G. Petersen, Jakob G. Simonsen, and Kasper Hornbæk. *Sketching Shape-Changing Interfaces: Exploring Vocabulary, Metaphors Use, and Affordances*, page 2740–2751. Association for Computing Machinery, New York, NY, USA, 2016.

[176] Michael Rees. From outsider to established-explaining the current popularity and acceptability of tattooing. *Historical Social Research/Historische Sozialforschung*, 41(3):157–174, 2016.

[177] Beate Reiser, Tobias Kraus, Lola Gonzalez-Garcia, Johannes H.M. Maurer, and Ioannis Kanelidis. Conductive nanocomposites, April 2017. Patent No. WO2017045989A1, Filed September 8th., 2016, Issued March 23th., 2017.

[178] Jun Rekimoto. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems Changing our world, changing ourselves - CHI '02*, page 113, New York, New York, USA, apr 2002. ACM Press.

[179] Alec Rivers, Ilan E. Moyer, and Frédo Durand. Position-correcting tools for 2d digital fabrication. *ACM Trans. Graph.*, 31(4), July 2012.

[180] Simon Robinson, Jennifer Pearson, Mark D. Holton, Shashank Ahire, and Matt Jones. Sustainabot - exploring the use of everyday foodstuffs as output and input for and with emergent users. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–12, New York, NY, USA, 2019. Association for Computing Machinery.

[181] Dusty Robotics. The world's leading robotic layout system, 2022. Accessed: 2022-11-22.

[182] Rugged Robotics. Building better, 2022. Accessed: 2022-11-22.

[183] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

[184] Thijs Roumen, Bastian Kruck, Tobias Dürschmid, Tobias Nack, and Patrick Baudisch. Mobile fabrication. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, page 3–14, New York, NY, USA, 2016. Association for Computing Machinery.

[185] Kimiko Ryokai, Stefan Marti, and Hiroshi Ishii. I/o brush: Drawing with everyday objects as ink. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, page 303–310, New York, NY, USA, 2004. Association for Computing Machinery.

[186] Jyoti Sachdeva. Valentine's day special arabic henna mehndi designs with cookie cutter, 2018. Accessed: 2020-09-01.

[187] Jogendra Saksena. Henna for happiness; india's "mehndi" art of symbols for all seasons. *The UNESCO Courier: a window open on the world*, XXX, 2:18–22, 1977.

[188] Munehiko Sato, Ivan Poupyrev, and Chris Harrison. Touché: Enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 483–492, New York, NY, USA, 2012. ACM.

[189] Valkyrie Savage, Xiaohan Zhang, and Björn Hartmann. Midas: Fabricating custom capacitive touch sensors to prototype interactive objects. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 579–588, New York, NY, USA, 2012. Association for Computing Machinery.

[190] Martin Schmitz, Mohammadreza Khalilbeigi, Matthias Balwierz, Roman Lissermann, Max Mühlhäuser, and Jürgen Steimle. Capricate: A fabrication pipeline to design and 3d print capacitive touch sensors for interactive objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pages 253–258, New York, NY, USA, 2015. ACM.

[191] Eldon Schoop, Michelle Nguyen, Daniel Lim, Valkyrie Savage, Sean Follmer, and Björn Hartmann. Drill sergeant: Supporting physical construction projects through an ecosystem of augmented tools. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '16, pages 1607–1614, New York, NY, USA, 2016. Association for Computing Machinery.

[192] Michael Schrage. Cultures of prototyping. *Bringing design to software*, 4(1):1–11, 1996.

[193] Gabriel Schwartz and Ko Nishino. Recognizing material properties from images. *CoRR*, abs/1801.03127:207–217, 2018.

[194] Evil Mad Scientist. Axidraw: Writing and drawing macines, 2022. Accessed: 2022-11-22.

[195] Circuit Scribe. https://circuitscribe.com/, 2020. Accessed: 2020-08-15.

[196] Ronald WB Scutt and Christopher Gotch. *Art, sex and symbol: The mystery of tattooing*. AS Barnes, South Brunswick, 1974.

[197] Ryan Seguine. Capacitive sensing techniques and considerations. Mobile Handset DesignLine, 2007. Accessed: 2018-09-20.

[198] Roy Shilkrot, Pattie Maes, Joseph A. Paradiso, and Amit Zoran. Augmented airbrush for computer aided painting (cap). *ACM Trans. Graph.*, 34(2), March 2015.

[199] Neo Smartpen. https://www.neosmartpen.com/, 2020. Accessed: 2020-02-15.

[200] Hyunyoung Song, Tovi Grossman, George Fitzmaurice, François Guimbretiere, Azam Khan, Ramtin Attar, and Gordon Kurtenbach. Penlight: Combining a mobile projector and a digital pen for dynamic visual overlay. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 143–152, New York, NY, USA, 2009. Association for Computing Machinery.

[201] Sparkfun. Force sensitive resistor - small, 2022. Accessed: 2022-06-02.

[202] Sparkfun. Force sensitive resistor - square, 2022. Accessed: 2022-06-02.

[203] J. Steimle, J. Bergstrom-Lehtovirta, M. Weigel, A. S. Nittala, S. Boring, A. Olwal, and K. Hornbæk. On-skin interaction using body landmarks. *Computer*, 50(10):19–27, 2017.

[204] Jürgen Steimle. Designing pen-and-paper user interfaces for interaction with documents. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, TEI '09, page 197–204, New York, NY, USA, 2009. Association for Computing Machinery.

[205] Jürgen Steimle, Mohammadreza Khalilbeigi, Max Mühlhäuser, and James D. Hollan. Physical and digital media usage patterns on interactive tabletop surfaces. In *ACM International Conference on Interactive Tabletops and Surfaces*, ITS '10, page 167–176, New York, NY, USA, 2010. Association for Computing Machinery.

[206] Evgeny Stemasov, Tobias Wagner, Jan Gugenheimer, and Enrico Rukzio. *Mix&Match: Towards Omitting Modelling Through In-Situ Remixing of Model Repository Artifacts in Mixed Reality*, page 1–12. Association for Computing Machinery, New York, NY, USA, 2020.

[207] Paul Strohmeier, Juan Pablo Carrascal, and Kasper Hornbæk. What can doodles on the arm teach us about on-body interaction? In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '16, pages 2726–2735, New York, NY, USA, 2016. ACM.

[208] Paul Strohmeier, Jarrod Knibbe, Sebastian Boring, and Kasper Hornbæk. Zpatch: Hybrid resistive/capacitive etextile input. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '18, page 188–198, New York, NY, USA, 2018. Association for Computing Machinery.

[209] Paul Strohmeier, Narjes Pourjafarian, Marion Koelle, Cedric Honnet, Bruno Fruchard, and Jürgen Steimle. Sketching on-body interactions using piezo-resistive kinesiology tape. In *Proceedings of the Augmented Humans International Conference*, AHs '20, New York, NY, USA, 2020. Association for Computing Machinery.

[210] Blair Subbaraman and Nadya Peek. P5.fab: Direct control of digital fabrication machines from a creative coding environment. In *Designing Interactive Systems Conference*, DIS '22, page 1148–1161, New York, NY, USA, 2022. Association for Computing Machinery.

[211] Zhenbang Sun, Changhu Wang, Liqing Zhang, and Lei Zhang. Query-adaptive shape topic mining for hand-drawn sketch recognition. In *Proceedings of the 20th ACM International Conference on Multimedia*, MM '12, page 519–528, New York, NY, USA, 2012. Association for Computing Machinery.

[212] Ivan E Sutherland. Sketchpad a man-machine graphical communication system. *Simulation*, 2(5):R–3, 1964.

[213] Ryo Suzuki, Rubaiat Habib Kazi, Li-yi Wei, Stephen DiVerdi, Wilmot Li, and Daniel Leithinger. *RealitySketch: Embedding Responsive Graphics and Visualizations in AR through Dynamic Sketching*, page 166–181. Association for Computing Machinery, New York, NY, USA, 2020.

[214] OmniVision Technologies. Ov5640 datasheet, 2011. Accessed: 2021-06-02.

[215] Rundong Tian and Eric Paulos. Adroid: Augmenting hands-on making with a collaborative robot. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, UIST '21, page 270–281, New York, NY, USA, 2021. Association for Computing Machinery.

[216] Iremnur Tokac, Benay Gursoy, Herman Bruyninckx, and Andrew Vande Moere. Craft-inspired digital fabrication: A study of interactive robotic clay carving. In *Proceedings of the 7th Annual ACM Symposium on Computational Fabrication*, SCF '22, New York, NY, USA, 2022. Association for Computing Machinery.

[217] Shaper Tools. Shaper origin, 2022. Accessed: 2022-11-22.

[218] Patrick Tresset and Frederic Fol Leymarie. Portrait drawing by paul the robot. *Computers & Graphics*, 37(5):348–363, 2013.

[219] Hannah Twigg-Smith, Jasper Tran O'Leary, and Nadya Peek. Tools, tricks, and hacks: Exploring novel digital fabrication workflows on #plottertwitter. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery.

[220] Virag Varga, Gergely Vakulya, Alanson Sample, and Thomas R. Gross. Enabling interactive infrastructure with body channel communication. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(4):169:1–169:29, January 2018.

[221] Manik Varma and Andrew Zisserman. A statistical approach to material classification using image patch exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(11):2032–2047, 2009.

[222] Katia Vega, Abel Arrieta, Felipe Esteves, and Hugo Fuks. Fx e-makeup for muscle based interaction. In Aaron Marcus, editor, *Design, User Experience, and Usability. User Experience Design for Everyday Life Applications and Services*, pages 643–652, Cham, 2014. Springer International Publishing.

[223] Katia Vega and Hugo Fuks. Beauty technology: Body surface computing. *Computer*, 47(4):71–75, April 2014.

[224] Katia Vega, Nan Jiang, Xin Liu, Viirj Kan, Nick Barry, Pattie Maes, Ali K. Yetisan, and Joe Paradiso. The dermal abyss: Color-changing tattoos for medical diagnostics. *GetMobile: Mobile Comp. and Comm.*, 22(2):38–42, September 2018.

[225] Artie Venti. How do tattoo artists transfer designs from drawing to skin?, 2019.

[226] Nicolas Villar, Daniel Cletheroe, Greg Saul, Christian Holz, Tim Regan, Oscar Salandin, Misha Sra, Hui-Shyong Yeo, William Field, and Haiyan Zhang. Project zanzibar: A portable and flexible tangible interaction platform. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–13, New York, NY, USA, 2018. Association for Computing Machinery.

[227] Wacom. Bamboo, 2022. Accessed: 2022-06-02.

[228] WallPen. Wallpen, 2017. Accessed: 2022-06-02.

[229] Beirong Wang, Jian Sun, and Beryl Plimmer. Exploring sketch beautification techniques. In *Proceedings of the 6th ACM SIGCHI New Zealand Chapter's International Conference on Computer-Human Interaction: Making CHI Natural*, CHINZ '05, page 15–16, New York, NY, USA, 2005. Association for Computing Machinery.

[230] Yanan Wang, Shijian Luo, Yujia Lu, Hebo Gong, Yexing Zhou, Shuai Liu, and Preben Hansen. Animskin: Fabricating epidermis with interactive, functional and aesthetic color animation. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, DIS '17, pages 397–401, New York, NY, USA, 2017. ACM.

[231] Keita Watanabe and Michiaki Yasumura. Flexiblebrush: A realistic brush stroke experience with a virtual nib. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, UIST '07, pages 47–48, New York, NY, USA, 2007. Association for Computing Machinery.

[232] Ian Watts. Red ochre, body painting, and language: interpreting the blombos ochre. *The cradle of language*, 2:93–129, 2009.

[233] Waveshare. 3.5inch rpi lcd, 2022. Accessed: 2022-06-02.

[234] Christian Weichel, John Hardy, Jason Alexander, and Hans Gellersen. Reform: Integrating physical and digital design through bidirectional fabrication. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, page 93–102, New York, NY, USA, 2015. Association for Computing Machinery.

[235] Christian Weichel, Manfred Lau, David Kim, Nicolas Villar, and Hans W. Gellersen. Mixfab: A mixed-reality environment for personal fabrication. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, page 3855–3864, New York, NY, USA, 2014. Association for Computing Machinery.

[236] Weicon. https://www.weicon.de/en/products/new-products/1219/isolation-spray, 2020. Accessed: 2021-06-15.

[237] Martin Weigel, Tong Lu, Gilles Bailly, Antti Oulasvirta, Carmel Majidi, and Jürgen Steimle. 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*, CHI '15, pages 2991–3000, New York, NY, USA, 2015. ACM.

[238] Martin Weigel, Vikram Mehta, and Jürgen Steimle. More than touch: Understanding how people use skin as an input surface for mobile computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 179–188, New York, NY, USA, 2014. Association for Computing Machinery.

[239] Martin Weigel, Aditya Shekhar Nittala, Alex Olwal, and Jürgen Steimle. Skinmarks: Enabling interactions on body landmarks using conformal skin electronics. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3095–3105, New York, NY, USA, 2017. Association for Computing Machinery.

[240] Michael Wessely, Ticha Sethapakdi, Carlos Castillo, Jackson C. Snowden, Ollie Hanton, Isabel P. S. Qamar, Mike Fraser, Anne Roudaut, and Stefanie Mueller. Sprayable User Interfaces: Prototyping Large-Scale Interactive Surfaces with Sensors and Displays. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–12, Honolulu HI USA, April 2020. ACM.

[241] Cliff White. *Classic Tattoo Stencils 2: More Designs in Acetate*. Number no. 2 in 1. Schiffer Publishing Limited, Pennsylvania, USA, 2016.

[242] Irmandy Wicaksono and Joseph A. Paradiso. Fabrickeyboard: multimodal textile sensate media as an expressive and deformable musical interface. In *New Interfaces for Musical Expression*, 2017.

[243] Karl D.D. Willis, Cheng Xu, Kuan-Ju Wu, Golan Levin, and Mark D. Gross. Interactive fabrication: New interfaces for digital fabrication. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '11, pages 69–72, New York, NY, USA, 2011. ACM.

[244] Andrew D. Wilson. Using a depth camera as a touch sensor. In *ACM International Conference on Interactive Tabletops and Surfaces*, ITS '10, pages 69–72, New York, NY, USA, 2010. ACM.

[245] Raphael Wimmer and Patrick Baudisch. Modular and deformable touch-sensitive surfaces based on time domain reflectometry. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 517–526, New York, NY, USA, 2011. ACM.

[246] Anusha Withana, Daniel Groeger, and Jürgen Steimle. Tacttoo: A thin and feel-through tattoo for on-skin tactile output. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18, pages 365–378, New York, NY, USA, 2018. ACM.

[247] Jacob O. Wobbrock, Andrew D. Wilson, and Yang Li. Gestures without libraries, toolkits or training: A $1 recognizer for user interface prototypes. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, UIST '07, page 159–168, New York, NY, USA, 2007. Association for Computing Machinery.

[248] Jie Wu, Changhu Wang, Liqing Zhang, and Yong Rui. Sketch recognition with natural correction and editing. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 951–957, New York, NY, USA, 2014. AAAI Press.

[249] Pan-Long Wu, Yi-Chin Hung, and Jin-Siang Shaw. Artistic robotic pencil sketching using closed-loop force control. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 236(17):9753–9762, 2022.

[250] Te-Yen Wu, Lu Tan, Yuji Zhang, Teddy Seyed, and Xing-Dong Yang. Capacitivo: Contact-Based Object Recognition on Interactive Fabrics using Capacitive Sensing. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pages 649–661, Virtual Event USA, October 2020. ACM.

[251] Jun Xie, Aaron Hertzmann, Wilmot Li, and Holger Winnemöller. Portraitsketch: Face sketching assistance for novices. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 407–417, New York, NY, USA, 2014. Association for Computing Machinery.

[252] Jun Xing, Li-Yi Wei, Takaaki Shiratori, and Koji Yatani. Autocomplete hand-drawn animations. *ACM Trans. Graph.*, 34(6), oct 2015.

[253] Junichi Yamaoka and Yasuaki Kakehi. Depend: Augmented handwriting system using ferromagnetism of a ballpoint pen. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 203–210, New York, NY, USA, 2013. Association for Computing Machinery.

[254] Mayu M. Yamashita, Junichi Yamaoka, and Yasuaki Kakehi. Enchanted scissors: A scissor interface for support in cutting and interactive fabrication. In *ACM SIGGRAPH 2013 Posters*, SIGGRAPH '13, New York, NY, USA, 2013. Association for Computing Machinery.

[255] Sang Ho Yoon, Siyuan Ma, Woo Suk Lee, Shantanu Thakurdesai, Di Sun, Flávio P. Ribeiro, and James D. Holbery. Hapsense: A soft haptic i/o device with uninterrupted dual functionalities of force sensing and vibrotactile actuation. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, pages 949–961, New York, NY, USA, 2019. Association for Computing Machinery.

[256] Chuang-Wen You, Min-Wei Hung, Ximeng Zhang, Po-Chun Huang, and Hsin-Liu (Cindy) Kao. Online survey study on social perceptions towards color-changing on-skin displays. In *Proceedings of the 2020 International Symposium on Wearable Computers*, ISWC '20, page 90–95, New York, NY, USA, 2020. Association for Computing Machinery.

[257] Chuang-Wen You, Ya-Fang Lin, Elle Luo, Hung-Yeh Lin, and Hsin-Liu (Cindy) Kao. Understanding social perceptions towards interacting with on-skin interfaces in public. In *Proceedings of the 23rd International Symposium on Wearable Computers*, ISWC '19, page 244–253, New York, NY, USA, 2019. Association for Computing Machinery.

[258] Ling Zhang, Hongjun Ji, Houbing Huang, Ning Yi, Xiaoming Shi, Senpei Xie, Yaoyin Li, Ziheng Ye, Pengdong Feng, Tiesong Lin, Xiangli Liu, Xuesong Leng, Mingyu Li, Jiaheng Zhang, Xing Ma, Peng He, Weiwei Zhao, and Huanyu Cheng. Wearable circuits sintered at room temperature directly on the skin surface for health monitoring. *ACS Applied Materials & Interfaces*, 12(40):45504–45515, 2020. PMID: 32911929.

[259] Yang Zhang, Gierad Laput, and Chris Harrison. Electrick: Low-cost touch sensing using electric field tomography. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 1–14, New York, NY, USA, 2017. ACM.

[260] Yang Zhang, Chouchang (Jack) Yang, Scott E. Hudson, Chris Harrison, and Alanson Sample. Wall++: Room-scale interactive and context-aware sensing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 273:1–273:15, New York, NY, USA, 2018. ACM.

[261] Zhijie Zhu, Shuang-Zhuang Guo, Tessa Hirdler, Cindy Eide, Xiaoxiao Fan, Jakub Tolar, and Michael C. McAlpine. 3d printed functional and biological materials on moving freeform surfaces. *Advanced Materials*, 30(23):1707495, 2018.

[262] Thomas G. Zimmerman, Joshua R. Smith, Joseph A. Paradiso, David Allport, and Neil Gershenfeld. Applying electric field sensing to human-computer interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pages 280–287, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[263] Amit Zoran. Hybrid craft: Showcase of physical and digital integration of design and craft skills. In *ACM SIGGRAPH Art Gallery*, SIGGRAPH '15, page 384–398, New York, NY, USA, 2015. Association for Computing Machinery.

[264] Amit Zoran and Joseph A. Paradiso. Freed: A freehand digital sculpting tool. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2613–2616, New York, NY, USA, 2013. ACM.

[265] Amit Zoran, Roy Shilkrot, Pragun Goyal, Pattie Maes, and Joseph A. Paradiso. The wise chisel: The rise of the smart handheld tool. *IEEE Pervasive Computing*, 13(3):48–57, 2014.

[266] Amit Zoran, Roy Shilkrot, and Joseph Paradiso. Human-computer interaction for hybrid carving. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 433–440, New York, NY, USA, 2013. Association for Computing Machinery.