Real-Time Capture of Holistic Tangible Interactions

Virág Varga
ETH Zurich &
Disney Research Studios
Zürich, Switzerland

Gergely Vakulya
ETH Zurich
Zürich, Switzerland

Benjamin Bürgisser
ETH Zurich
Zürich, Switzerland

Nathan Riopelle
University of Michigan
Ann Arbor, MI, USA

Fabio Zünd
ETH Zurich
Zürich, Switzerland

Robert W. Sumner
Disney Research Studios
Zürich, Switzerland

Thomas R. Gross
ETH Zurich
Zürich, Switzerland

Alanson P. Sample
University of Michigan
Ann Arbor, MI, USA

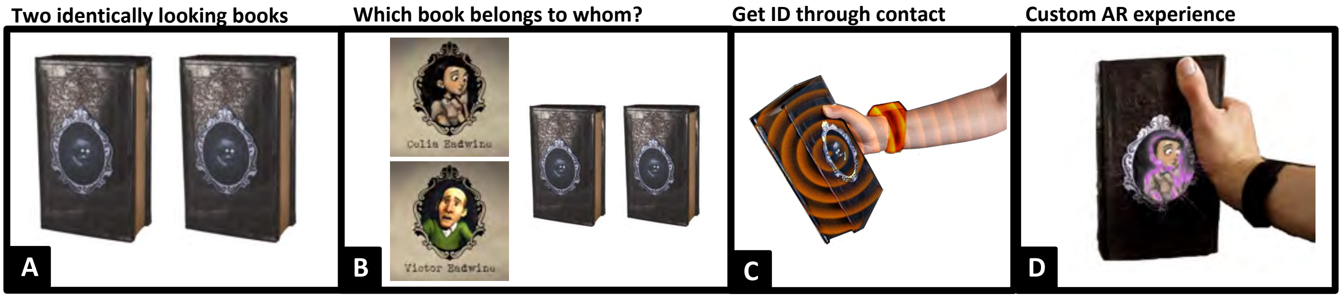


Figure 1: Example use case supported by TangibleID: Differentiating between identically looking objects is possible by retrieving the unique ID of the object at the moment of physical contact. Identity and time allow enriching immersive applications.

ABSTRACT

When digital applications aim to blend virtual and real worlds, understanding the actual physical actions of users becomes an important task; the precise timing of these tangible interaction events is needed, along with the identity, and possibly location and history, of all involved actors/objects. With multiple actors or objects, it is difficult to identify who touches which object and when. Instrumenting objects for Body Channel Communication (BCC) allows message exchange around the human body between instrumented objects and the user themselves. In this paper we show how BCC can be utilized to perform under real-time conditions so that we can directly notice touch events (and the identity of actors). TangibleID is a framework that unifies tangible interaction capture for objects and users based on wearable BCC. TangibleID provides identification and communication with tagged objects/users in less than 120 ms and supports a variety of tangible interactions, without the need to restrict user (hand) movements or to maintain line-of-sight connection to cameras. When an AR application is combined with TangibleID, a new tangible mixed reality experience is achieved, as demonstrated in the "Haunted Castle" showcase. The paper presents an end-to-end technical evaluation including trade-offs regarding robustness and speed of touch recognition, outlines the breadth of interaction modalities, and reports on an initial user assessment.

CCS CONCEPTS

• **Computer systems organization → Embedded systems**; • **Human-centered computing → Ubiquitous and mobile computing systems and tools**; *Mixed / augmented reality.*

KEYWORDS

System Design; Body Channel Communication; Tangible Interfaces; Interaction Capture; Augmented Reality

ACM Reference Format:
Virág Varga, Gergely Vakulya, Benjamin Bürgisser, Nathan Riopelle, Fabio Zünd, Robert W. Sumner, Thomas R. Gross, and Alanson P. Sample. 2021. Real-Time Capture of Holistic Tangible Interactions. In *Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '21), February 14–17, 2021, Salzburg, Austria.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3430524.3440658

1 INTRODUCTION

As applications move beyond the confinement of platforms such as smartphones or computers, they attempt to merge the real world and virtual worlds to create a meaningful co-existence of digital content with the surrounding environment. One important aspect of creating these immersive applications is incorporating physical objects, people, and events into the experience. Accordingly, research efforts have started to focus on creating more natural and more intuitive interaction modalities [9] – often by adapting

---

TEI '21, February14–17, 2021, Salzburg, Austria

ACM ISBN 978-1-4503-8213-7/21/02…$15.00
https://doi.org/10.1145/3430524.3440658

novel input technologies and by experimenting with 3D user interfaces [12, 27, 49], natural user interfaces [21, 35, 41], multi-modal interactions [15, 39, 40], and tangible user interfaces [48].

One of the key issues of capturing and "processing" the real world is to identify the objects or persons a user touches. 'Touch" can be defined in various ways: grasping an object, picking up one of several identical-looking objects, contact with other users, or being present at a specific location. Many systems include cameras that provide the overall context. As successful as computer vision (CV) has been, cameras often miss details related to the objects that users interact with, how those objects are used, as well as invisible events that occur outside the field of view of the camera or that are occluded by the user's actions. RFID tags allow the cheap instrumentation of many objects, and with appropriate tracking, the object movement may be related to human activity. But the role of objects with RFID tags is limited – they cannot capture new context and provide limited assistance in user localization (e.g., if there is no tag movement). Thus, it is desirable to look at alternative mechanisms for incorporating the real world into the virtual one, in a way that complements the capabilities of existing systems.

Traditional approaches to capture interactions (like CV or RFID) are approximate, they infer contact between objects, persons, and/or locations. Body Channel Communication (BCC) allows direct detection, as BCC relies on augmenting human touch with digital information through embedding data into an electric field that is then carried by the human body [57]. A BCC-user can exchange data with BCC-enabled objects or other users through physical contact. BCC has been demonstrated to work across the entire body allowing the system to detect a wide range of physical contacts that naturally occur [52, 53]. As BCC-enabled objects can maintain state (ID, history), it is possible to distinguish between two objects that look the same but have different properties (as seen in Figure 1).

TangibleID is a framework based on BCC that provides robust real-time interaction capture for end-user applications. Using TangibleID, applications can achieve instant bi-directional object-, person-, or location-recognition, which then opens up a series of novel storytelling opportunities. This paper discusses the overall system design, including physical prototyping and network protocols, and analyzes the performance that can be expected by applications.

Once the technical challenges of interaction capture are resolved, we then explore how integrating tangible interactions can create more immersive experiences. Augmented Reality (AR) is an important class of digital applications that are set in the physical world. AR systems have become effective at room level mapping and self-localization [16, 28, 56], but current AR experiences may leave users disconnected from the physical world, forcing them to *watch* rather than *participate* in the augmented environment [46]. With TangibleID, AR applications gain contextual awareness of when and how users interact with and manipulate tangible objects (see Figure 2). This capability offers users a more natural and intuitive way to interact with augmented environments, while also allowing virtual content to be overlaid in a robust and consistent manner.

AR applications can benefit from precise timing information about touch events. Azuma suggests three requirements that AR systems must meet [7]: (1) combination of real and virtual contents, (2) interactivity in real-time, and (3) registration in 3D. The first and third requirements must be dealt with by the AR core application itself. But the second requirement depends to a large extent on the framework that connects an AR application to objects and locations in the physical world. This second requirement then raises a series of questions: (i) What interaction modalities can be identified for the complete systems (ranging from tangible interaction capture to event processing); (ii) How can a tangible interaction capture framework based on BCC be added into an AR application while respecting the real-time constraints; and (iii) How can the physical integration of BCC (with AR, and into everyday objects) be realized? This paper does not only tackle these issues, but it also presents a complete application showcase as a real life demonstration of the systems capabilities. While the Haunted Castle leverages TangibleID specifically in an AR setting, one can easily imagine other end-user applications as well.

Contributions

The key contributions are summarized as follows:

- architecture (describing physical integration, software components, their relations, and revised networking protocols) to go from a communication infrastructure (based on BCC) to robust and scalable real-time tangible interaction recognition;
- the TangibleID system that implements the above architecture and its evaluation with regard to real-time constraints (capture to end-user application delivery in less than 120 ms) and design tradeoffs (ID beaconing pace, touch event recognition rate, and fairness in the presence of concurrent touch events);
- demonstration of the potential of TangibleID (and similar systems) for application development by identifying interaction modalities, and coupling them to an AR application for a complete showcase (the *Haunted Castle*) with preliminary end-user evaluation.

The paper serves as detailed integration guide for the construction of interactive systems that (i) require real-time interaction capture, (ii) want to unify handling of locations, users, and objects, and (iii) want to use an object's state to distinguish between identically looking objects.

2 RELATED WORK

The following section provides context in relationship to existing work in the areas of body channel communication and object interaction detection. As augmented reality is one of the prime application areas of our technology, we also overview how tangible interactions fit into the landscape of AR.

2.1 State-of-the-art body channel communication

Body Channel Communication (BCC) technology can enable non-traditional tangible interfaces, where physical objects with embedded BCC transceivers can communicate data back and forth with a wearable BCC transceiver through touch interaction. BCC enhanced objects transmit and receive weak electric signals (in the form of electric fields) that couple to the user's hand and propagate along the skin, arm [57], and eventually the whole body [53].

Arbitrary digital data can be embedded in this underlying electric signal, making the human body an isolated transmission path in this communication network.

Recently BCC has been gaining more attention and was proposed for identification [22, 23, 54], short range object recognition [18], and gaming [51], while wearable BCC devices now allow the construction of a flexible infrastructure [52]. However, an important step is still missing to couple BCC to application development so that new interactive experiences are possible. This paper provides a comprehensive approach showing how a BCC messaging framework can be leveraged for tangible interactions. TangibleID supports messages from BCC-enabled devices to an application to identify objects, users, and locations, as well as messages from an application to those devices to store data or to trigger haptic actions. This topic has not been addressed by earlier work [52] that focused on BCC as a bulk data transport mechanism and did not consider the timing implications of coupling BCC with interactive applications.

2.2 Understanding interactions through object recognition

There exist a handful of technologies that can possibly capture a wide range of user interactions that naturally occur in the everyday life. Table 1 compares various possibilities to identify their unique feature set.

Recent advancements in CV make it possible to achieve not only on-the-fly mapping of the environment, but object- and even touch recognition from the moving egocentric camera viewpoint, without imposing additional instrumentation burden on the environment (e.g., [25]). However, in such cases the user interactions are limited to grasping with the hands while line-of-sight contact must be guaranteed as well. Moreover, the tracking over a longer period of time might be disrupted by too much movement. External cameras (to provide a third-person viewpoint) may better cover the space but usually require substantial infrastructure investment. Furthermore, once several interactions can take place at once, the computational demands might incur prohibitive costs or violate the real-time constraints of an application. User differentiation might also be difficult as it would require preceding training and face recognition, let alone needing clear line of sight to the user’s face and also to their entire body to allow any body part to take part in the interaction.

Capacitive, acoustic, electromagnetic, and electromyographic sensing solutions (e.g., [17, 19, 30, 31, 47]) have the advantage that the objects to be tracked do not need to be instrumented. However, these approaches heavily rely on (often repeated) training, and struggle with interference once several actions take place at once. Moreover, given the usually weak signals and the sensing mechanism behind these techniques, the interactions are most often restricted, e.g., a specific hand (which has been instrumented) must be employed.

Another option is to instrument the environment with RFID readers and the objects with RFID tags and then to deduce interaction based on the movement of the tags. If, in addition to objects, the user is also instrumented, then it is possible to correlate user interaction with object contact (e.g., [37]). The low cost of the RFID tags makes this approach attractive if a large number of objects must be instrumented, but so far, support of multiple concurrent users has not been demonstrated, and it is not obvious how the approach can be extended to capture interaction between users. Furthermore, RFID tags provide read-only data, imposing restrictions on application development.

Table 1: High-level comparison of various object recognition techniques.

✅ Directly supported ❗ Supported with limitations ❌ Not supported Feature, functionality	Computer Vision (egocentric)	Computer Vision (third-person view)	Capacitive, EM, or Acoustic Sensing	IMU / Accelerometer	RFID	Body Channel Communication (in **TangibleID**)
Object recognition, without object instrumentation	✅	✅	✅	❌	❌	❌
Object recognition, without instrumentation of the environment	✅	❌	✅	✅	❌	❗
Robust object recognition, w/o training	❌	❌	❌	✅	❗	✅
Object recognition through contact with arbitrary body part	❗	❗	❗	✅	❌	✅
Works without line of sight	❌	❌	✅	✅	✅	✅
Unique objects (same look, individual ID/history, w/o temporal tracking)	❌	❌	❗	✅	✅	✅
Unique users (w/o training and/or temporal tracking)	✅	❗	❌	❌	❌	✅
Scalable interactions (number of parallel events or users)	❗	❗	❌	❗	❗	✅
Distinguishes between locations	✅	✅	❗	❌	❌	✅

Attaching IMUs (inertial measurement units) to the objects an application is interested in might seem to be a straightforward, low cost solution for object/interaction recognition (e.g., [55]). However, this approach may imply serious limitations for applications. IMU-based systems recognize movement so they can be used to track moving objects, but IMUs are not suitable for static physical contacts or for location identification. The biggest limitation, however, is that the interacting users cannot be differentiated, therefore no connection can be established between users and objects.

BCC (like capacitive sensing) uses electric fields to intercept the moment of touch (e.g., [18, 52]). While BCC imposes some instrumentation burden, the context-awareness it provides is user-centered, instant, robust (without any training), allows interaction capture for any body part, and scales for a number of simultaneous interactions by the same or by several users without increasing computational complexity. BCC can also distinguish between instances of the same object class, without the need of maintaining any temporal tracking. Moreover, BCC allows user IDs as well, both for capturing interpersonal interactions between users or for common access towards the same object by several users.

Table 1 compares the feature set of various tangible interaction capture possibilities. There might be several other dimensions along which object and consequently interaction recognition techniques could be compared against. For applications (e.g., AR) that are subject to real-time constraints or that include interaction with the environment (either to obtain context-specific information or to trigger context-specific actions like haptic feedback), a framework like TangibleID that is based on BCC offers a unique feature set that allows for a great variety of interactions.

2.3 Physical interactions in augmented reality

Tangible user interfaces (TUI) offer interactions with physical objects to access and manipulate digital data [24]. In fact, TUI and AR share the same vision of *embedding computing in existing environments and human practices to enable fluid transitions between the 'digital' and 'the real'* [48]. Hence the concept of Tangible AR (TAR) is a natural encounter of these two fields [10, 26], where physical manipulation can directly impact the rendered digital content. The typical interaction design in TAR applications usually covers tasks such as viewpoint control, selection & release, 3D manipulation, and event generation & system commands [32]. The proposed interactions include picking up objects, performing gestures with props, using a keyboard & mouse, pointing with bare hands, or using special devices. Most TAR projects solely rely on CV techniques to track the fiducial or object AR markers in space [11, 20, 33, 36, 42, 44]: the orientation, rotation, tilt, or partial occlusion (touch) of the physical AR marker is used to manipulate the displayed virtual content. The implementation sometimes also uses haptic [4, 8, 13, 29] or tactile devices [5].

Another aspect of physical interactions is the possibility of collaboration between multiple users. While only a handful of efforts focus on this aspect, they all only support mediated collaboration, when several users can watch and manipulate digital objects in the shared space [6, 13, 29, 34, 45]. However, direct interpersonal interactions – such as handshakes, high fives, etc. – between the users are not captured.

TangibleID supports both aspects. On one hand, we build on top of a traditional AR application by also supporting several TAR task targets, such as *selection & release* through simply making physical contact, while also allowing a unique approach to *event generation & system commands* through its in-built communication and ID-recognition abilities associated with the BCC backend. Moreover, a BCC interaction capture can also handle mediated (through an object) and direct interactions between users.

3 APPLICATION WORKFLOW WITH TANGIBLEID

The TangibleID system is comprised of several interconnected components, which allow for the detection of interaction events in the real world and then for the usage of those events in the end-user application (e.g., a game engine that interprets the touch data and then renders the appropriate graphics on a display). While the subsequent sections go through the details of all important system components, here we discuss the workflow of a possible application that is set in the augmented reality context, as seen in Figure 2. This example illustrates how well the TangibleID system can be combined with an existing application and the novel experience it enables. AR, however, is only one example application category that can benefit from TangibleID [50]. Other applications using different visual, audio, etc. feedback mechanisms, or simply with the intention of only recording, would follow a similar workflow.

Figure 2/A shows a user wearing a head-mounted display, here in the form of a Microsoft HoloLens, which allows her hands to be free for more natural and richer interactions with the physical world. The system could also be implemented on Android tablets as well, which allow for a *virtual window* AR experience. Both of these AR systems allow the scanning of the physical environment for graphical markers to best tailor rich visual media overlay on the real world.

Along with the AR display, the user also wears a BCC transceiver (as a wristband) that is wirelessly paired to the AR display. As discussed in Section 4.2, to increase user convenience, the BCC transceiver might be physically integrated either into the headset or into the tablet. The BCC transceiver transmits and receives signals that travel along the user's entire body, allowing for bi-directional communication when contact is made with tangible objects or the enhanced infrastructure containing BCC nodes. This can be seen in Figure 2/B, where the BCC wristband communicates with a BCC enhanced tangible in the form of an *enchanted book*. Tangibles can take nearly any physical form that allows for the integration of the newly designed TouchCom-Mini boards, which are described in Section 4.1. These boards offer a number of enhancements over previous BCC prototypes, including a 4x reduction in size, haptic feedback, and an on-board IMU.

Figure 2/C refers to TangibleID's ecosystem that enables tangible interaction capture on the application level. When a user touches a tangible device, the user's BCC transceiver implements a low latency discovery protocol to determine the identity, functional capabilities, and state of the TangibleID object. The interaction event is then passed to the AR application, which takes appropriate actions depending on the design of the AR experience. The visual data is fused with the BCC touch data, which does not only provide precise timing of touch events but can also ensure a more constant and reliable user experience in case objects are lost by visual tracking but are still visible via the BCC connection (see Section 8.1). Additionally, to monitor system performance for research purposes as well as to aid in the configuration, testing, and development of interactive games, a component called *Overwatch* has been implemented (see Section 5.1). Although not needed or intended for end user deployment of TangibleID, Overwatch offers a valuable tool to allow developers to monitor the system and push configuration and content updates to the BCC nodes via WiFi.

Figure 2/D shows the resulting user experience as seen through a HoloLens. When the user picks up the book, the BCC subsystem identifies the object and its capabilities. The action of picking up the BCC-*enhanced bottle* in the other hand causes the game engine to render graphics of *energy* moving from the book to the bottle. At the same time, the game engine sends haptic commands to the tangibles to initiate variable tactile feedback to correspond with the visuals. The result is a seamless user experience with the virtual content while also fitting more naturally into the real world.

4 PHYSICAL INTEGRATION

One of the key questions when developing TangibleID, and later integrating it into an end application, is how to instrument the various tangible objects, locations, and users. This section provides a discussion on this subject.

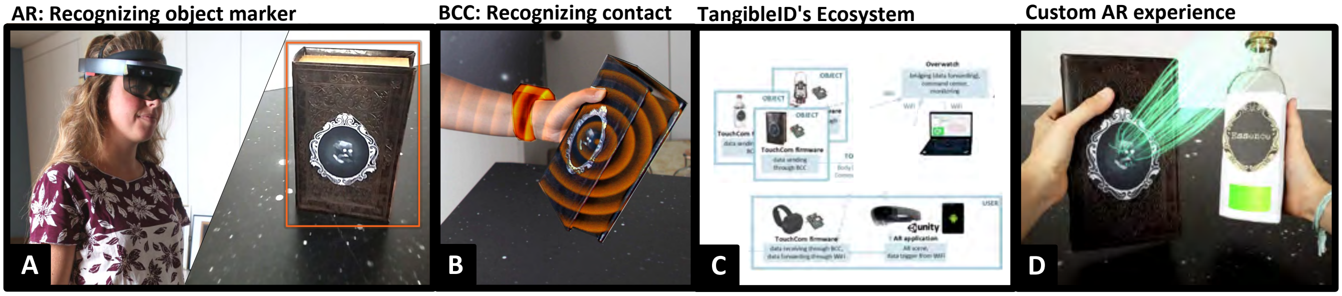


Figure 2: Integrating tangible interaction capture enables enriched immersive experiences, as illustrated in the context af an AR application: A) The user's AR glasses recognize the object's AR marker; B) Body Channel Communication recognizes the exact moment when an object is being touched and also extracts the object's ID with that same touch event; C) The touch information is forwarded to the AR application through a software ecosystem; D) The identified touch event triggers a custom experience in the AR story.

4.1 Tangibles

Wearable BCC units have traditionally been difficult to implement [19], but a recent design, TouchCom [50, 52], has demonstrated promising results with integrating the same hardware/software prototype in fixed infrastructure (floor tiles), portable objects, as well as in wearable devices. Moving from proof-of-concept BCC demonstrators to real life objects, however, is a non-trivial step.

4.1.1 *TouchCom-Mini.*

While TouchCom is effective as a proof-of-concept for a groundless BCC system, its large size (7 x 8 cm^2) and large external battery make it cumbersome to wear and integrate into everyday objects. Based on the original design, we developed a compact version. *TouchCom-Mini* consists of two stacked PCB boards, powered by a single cell battery (4.2 V); the dimensions are now 2.5 x 3.5 cm^2. The analog board is a simplified version of the original TouchCom circuit, offering only 4 MHz or 8 MHz as fixed carrier frequencies (with a peak-to-peak output voltage of 6.4 V and 6.7 V respectively). The digital board contains an STM32F415 microcontroller, with an additional surface mounted WiFi chip and a triaxial, low-g accelerometer sensor attached to it. Moreover, a coin vibration motor is connected to the board, to have the option to provide tactile feedback. Throughout Section 8.2) we use the BCC floor tiles as seen in the original TouchCom work [52]. However, all other tangibles are equipped with TouchCom-Mini, including a new, lightweight wristband. Moreover, the evaluation in Section 6 also uses TouchCom-Mini devices. Figure 3 reveals the details of one of the real life objects employed in the Haunted Castle.

4.1.2 *BCC electrode layout design.*

When turning an object into a BCC-enabled device, beyond mounting a TouchCom-Mini node, a proper electrode layout must be chosen that covers the possible touch points of the object well enough. BCC works over weak electric fields, and two electrodes (conductive, though might be insulated) are used to make sure the device can couple both to the environment (through the ground electrode) and to the user's body (through the signal electrode). The receiver analog circuit is designed to measure the electric potential difference at the two electrodes; this difference then is forwarded to the digital domain in the form of sampled ADC (analog-to-digital) values. When covering an object with BCC electrodes, it is important to ensure that the two types of electrodes are physically separated enough, to maximize the opportunity for them to reach different potentials (i.e., resulting in higher signal levels). The separation can be in the form of physical distance, e.g., by placing the electrodes vertically (with some distance) or horizontally next to each other – ensuring that the user comes into contact with only the signal electrode [53]. When the exact location of the touch contact on the object is not predictable – e.g., when covering a portable, 3D object – the optimal separation of the ground and signal electrodes is non-trivial. As a rule of thumb, an asymmetric pattern is recommended, to maximize the occurring potential difference. Experimentation and fine tuning can take place, e.g., using offline tools.

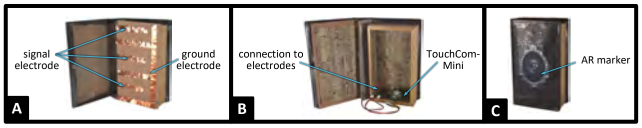


Figure 3: Example tangible construct. A: Right under the book cover, a set of electrodes are inserted that span across the whole book to support touch at any location. The signal and ground electrodes are separated from each other in an intersecting manner. B: Inside the book, there is a TouchCom-Mini device with optional network access for monitoring. C: The closed book has a visible AR marker (optional).

4.2 User-worn BCC device options

In most use cases of TangibleID (see Section 7 for details), the users need to be instrumented. There are several options for user-worn BCC device placements. The simplest option is a stand-alone BCC wristband (Figure 4/A). In case the user is equipped with additional devices (such as a tablet or HMD) as required by the end-user application, semantic pairing of these devices is possible, as later discussed in Section 5.2.

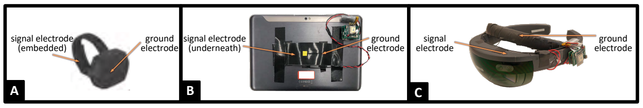


Figure 4: User-worn BCC device options. A: Light-weight BCC wristband. B: BCC-Tablet integration. C: BCC-HoloLens integration.

Beyond securing the data flow between different user devices (such as the BCC identifier of the user and the user's tablet/HMD), physical integration can be explored as well to increase convenience. If smartphones or tablets are used as a display, a BCC-tagged case could naturally couple the user to the BCC sensor as they hold the device. Figure 4/B shows a possible design. While its communication performance is slightly weaker than the wristband version, it still offers sufficient signal level to guarantee successful data transfer. When an HMD is used as display, the integration is more challenging, since the BCC signal tends to poorly propagate around the head [53]. Moreover, hair tends to degrade the coupling property. Nonetheless, the BCC-HMD design depicted in Figure 4/C matches the performance of the tablet design.

To ensure BCC devices meet established safety limits, we used a Narda NBM-555 Broadband Field Meter with a EF-0392 Electric Field Probe to measure the electric field generated by each device. According to the FCC Guidelines for Human Exposure to Radio Frequency EM Fields [14], the limit for the Maximum Permissible Exposure for the General Population / Uncontrolled Exposure in this frequency band is 103 V/m. The peak measured E-field valued for the BCC wristband (2.40 V/m), the HoloLens with integrated BCC module (2.03 V/m), and the tablet with integrated BCC module (2.72 V/m) are all well below the exposure limits and less than, or equivalent, to typical electronics found in the home and office.

5 SOFTWARE ECOSYSTEM DESIGN

Several standalone software components must cooperate for a seamless experience when the TangibleID framework is combined with an application [50]. Figure 6 depicts the system components and their relation. The two eassential subsystems in the ecosystem are:

1. the *embedded BCC firmware* that runs on the BCC nodes (embedded in tangibles, in the infrastructure, or user-worn). It connects BCC endpoints using established network protocols and has a data forwarding capability (incoming BCC data through an outgoing data bridge); and
2. the *end-user application* that runs on a mobile device (i.e., HMD, tablet, or smartphone). It has an open touch input channel (through an incoming data bridge), and may or may not use other information for its operation as well (such as visual discovery and display).

In this section we describe how the TangibleID network can be monitored, and we also discuss how to resolve data bridging between devices that semantically belong together.

5.1 Overwatch

To support the research and development efforts of the project, we developed a tool called *Overwatch*. It is implemented as a multi-threaded TCP server application with a rich UI, written for desktop platforms using C++ and the Qt framework [3]. The feature set of Overwatch makes it a crucial tool for TangibleID application development, while it is also optional for deployment. All devices that are part of the TangibleID network, such as all BCC nodes and mobile devices that run the end-user applications, are programmed to log in over WiFi to Overwatch's IP address by default. Through the *Network overview* we can list all connected devices while the *Network command center* allows controlling all BCC nodes remotely, without needing to re-flash the firmware on the devices to change functionality. Additional functionalities allow real-time plotting of the RX performance (*Aggregated view* and *Raw value debugger*). Figure 5 shows a screenshot of the running application.

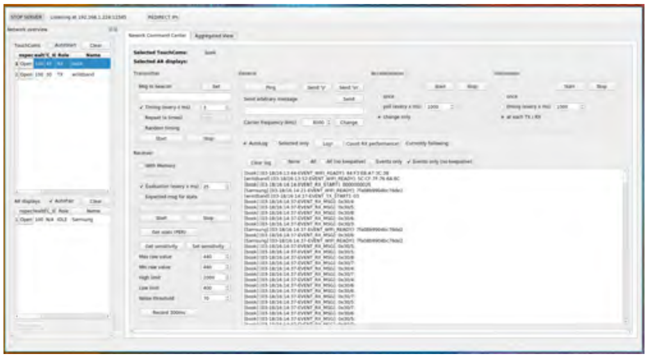


Figure 5: Screenshot of *Overwatch*. Its purpose is to monitor and manage the whole TangibleID network, and to offer functionality to connect those devices to end-user applications (that may be run on a tablet or HMD) as well.

5.2 Data bridging topologies for user-worn devices

Applications that need tangible capture usually already equip their users with a mobile device, such as a tablet or HMD. With TangibleID, the user also carries a BCC device. Regardless if these user-carried devices are physically co-located or are separate (as discussed in Section 4.2), the data flow must be guaranteed.

During implementation and evaluation of TangibleID applications, we use Overwatch to continuously monitor the system performance. Furthermore, since the WiFi backbone connection is already switched on to enable monitoring, we simply employ Overwatch also as data bridge. This setup results in a star topology, where Overwatch is the central server, and all devices are programmed to register themselves with it. Overwatch then takes care of automatic message forwarding between the linked mobile devices (tablet, smartphone, HMD) and BCC devices that belong together – without modifying the stream content. This data bridging topology guarantees complete access for developers to data traffic.

Real-time high temporal resolution for touch recognition is a desired property in a TangibleID application. High resolution can be achieved through frequent touch sampling over BCC. However, it is non-trivial to decide how to propagate that information to the end-user application when (near) real-time system reaction is

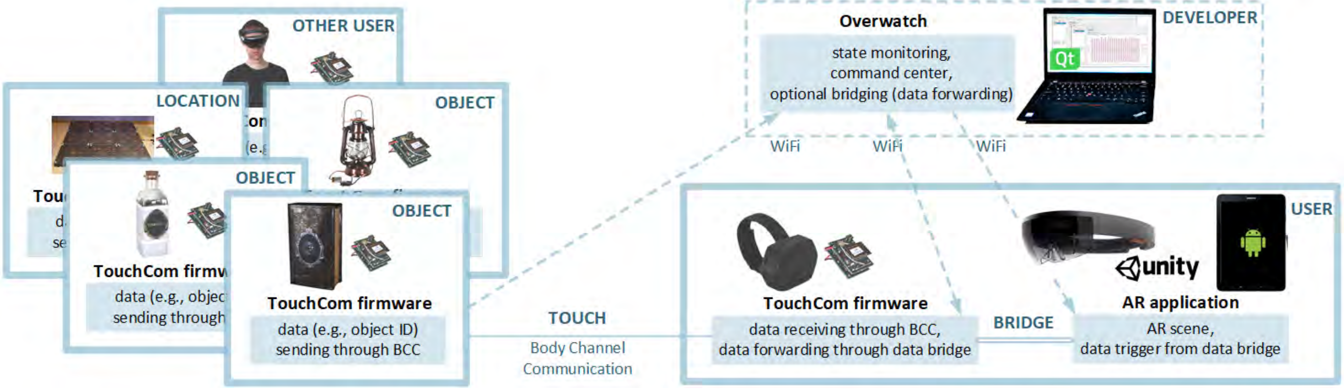


Figure 6: Overview of the ecosystem in the TangibleID framework. The physical interaction is captured through BCC, the data is then forwarded through a bridging mechanism from the TouchCom node to the end-user application (AR or otherwise); *Overwatch* optionally monitors the nodes' states through WiFi.

desired: heavy TCP traffic can easily lead to delayed data processing, resulting in a lagged experience – as later seen in our best vs worst case end-to-end reaction time. To address this issue alternative bridging mechanisms may be considered. A direct physical connection between mobile devices and BCC devices, e.g., with direct wiring, is one option. TouchCom-Mini supports USB, therefore it could easily be plugged into an Android device. This setup removes any networking overhead, but the USB connection and processing can still cause a non-negligible delay. Moreover, HMDs like the HoloLens do not support easy hardware extension. Another option is a mesh topology based on WiFi UDP connections between BCC and mobile devices. UDP is not a reliable protocol but has less overhead than TCP and only delivers *fresh* messages. With UDP multicast, Overwatch can still monitor traffic without adding extra overhead. A third alternative that not only decreases networking traffic but also reduces processing overhead is to propagate only *touch event changes* towards the game engine (e.g., `onTouch` and `onRelease`).

6 INTERACTION CAPTURE AND EVALUATION

This section explores how the BCC networking can be reconfigured to support the real-time constraint of tangible interactions. Additionally, the expected performance of an application that adapts TangibleID for interaction capture is also investigated. We focus on four main aspects: (i) achievable throughput for fast touch discovery on the BCC level; (ii) how that throughput translates to system performance on the software ecosystem level, i.e., how fast can we process a single touch event; (iii) how fast can multiple touch events be handled; and (iv) an end-to-end evaluation of touch events: what is the accuracy and latency experienced by the user at the application level.

6.1 Throughput on the BCC level

Most BCC efforts focus on general purpose data transfer. In comparison, TangibleID employs BCC primarily to recognize – as fast as possible – when touch events occur and to identify the touched objects' IDs. Hence, here we need to try to maximize the number of BCC packets going through. While TXs are programmed to beacon their IDs regularly – which then are picked up by RXs – we want to ensure that beaconing can be repeated as often as possible. We program the TouchCom devices with a firmware that supports a shortened (ID-)packet type as well. The packet structure can be seen in Figure 7: the 6 bytes in total, with 1 byte of data, is transmitted in 2.19 ms; with a 0.70 ms receiver processing delay, it takes a total of 2.89 ms. This performance is approx. 2.2x speed up compared to the original TouchCom messaging performance.

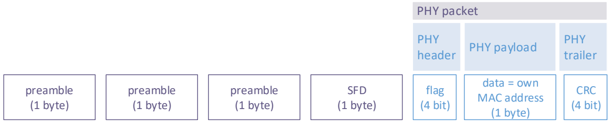


Figure 7: ID-type packet structure in TouchCom's packets.

When BCC is used as one component of a bigger system, various factors and their tradeoffs might impact the BCC performance. One of the emergent properties we found while building an end-user application with TangibleID is that WiFi traffic on the TouchCom-Mini affects the BCC channel. While WiFi and BCC both use EM signals, they operate on different frequencies (2.4 GHz vs 8 MHz), therefore, using both technologies simultaneously should be free of interference. However, WiFi traffic causes visible noise peaks in the sampled TouchCom signal. The more intense the WiFi traffic, the more noise is imposed on the BCC channel. The principal suspected sources for this interference are the processing overhead and power supply noise caused by the heavy WiFi usage. To understand the extent of this interference, we evaluated a TX beaconing pace of 3/5/15/25/50/100 ms; in each case the BCC TX transmits a total of 10000 ID-packets to a BCC RX every x ms. Figure 8 shows that when the WiFi is switched off at the RX, 95+% of the packets sent by TX correctly arrive at RX. However, once the WiFi is switched on, and continuous data forwarding to Overwatch is enabled, we measure only 88-82% at a pace of 3 or 5 ms, with the performance reaching 95+% only when the TX beaconing pace is 25 ms or more.

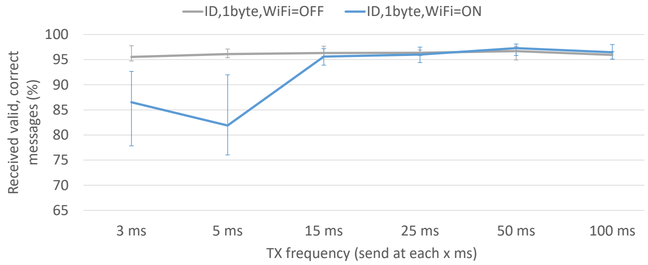


Figure 8: Beaconing pace performance for various configurations. If immediate WiFi data forwarding is switched on the RX, the BCC performance degrades.

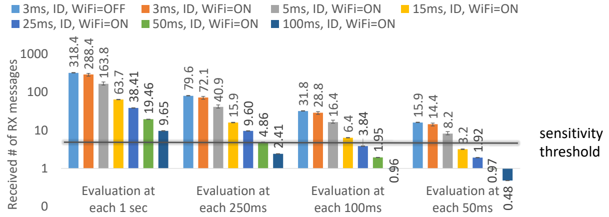


Figure 9: Temporal touch resolution using different beaconing strategies (logarithmic scale). The results are presented in 4 different ways, where the size of the evaluation window varies between 50 ms and 1 sec.

6.2 Temporal resolution for touch recognition

To understand what the BCC throughput means in terms of application development when BCC is employed as a touch recognition device, we translate the network performance to a meaningful application-level metric. The temporal resolution shows how fast/how often the system can recognize touch events. Figure 9 depicts the possible temporal touch resolutions as perceived at Overwatch or the end-user application (e.g., a game engine), under different beaconing strategies – grouped by different evaluation time window preferences. If the BCC protocol on the TX is configured to beacon every 3 ms and WiFi data forwarding is switched on, then on average 288 touch events are correctly received in a second, or 14 events if evaluated every 50 ms. Although performance degrades at a fast beaconing pace, the system throughput (i.e., correctly received packets per time window) is still higher than at a lower beaconing pace.

The data shown in Figure 9 only count correctly received messages (i.e., the received message was the one beaconing by the TX), however, in each case a few additional packets (0-1.5%) showed up at the RX, flagged by the firmware as correct, due to imperfections of the chosen error detection approach (4 bit CRC on 1 byte data). To filter out these unnoticed errors and to deal with *cross talk* that sometimes appears due to movement or when objects are located too close to each other[1], a *touch sensitivity threshold* can be defined. According to this touch sensitivity threshold, only identified touch events that occur at least X times in a given window should be forwarded to the end-user application. X should be chosen based on the expected number of packets per evaluation window. Figure 9 illustrates the threshold being set to 5, meaning that 5 RX messages are required per evaluation window from any BCC transmitter in order to the application be notified about the touch occurring.

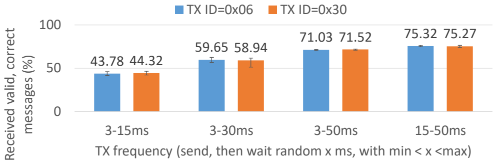


Figure 10: Two TX simultaneously sending to one RX. Both TX is configured to send their IDs, then back off for random amount of time. The plots shows the success rate of the two IDs going though, under 4 random backoff strategies, proving that fairness is guaranteed in all cases.

6.3 Recognizing concurrent touch events

When using several TouchCom TX devices at the same time, we encounter the challenge of multiple access over the same medium. While established networking protocols could be applied to guarantee deterministic access for all competing nodes, the runtime overhead of those techniques would be counter-effective while trying to use BCC as an instant touch recognition (and identification) mechanism. Therefore, instead of establishing dedicated connections, we aim to maximize beaconing throughput. In the new firmware, we use an optimized version of the RWB (Random Waittime Beaconing) protocol [52], employing not only the shortened ID-type packets, but exploring different waittime configurations.

First, we evaluate the *fairness* of such a protocol: given two TXs, i.e., when recognizing two concurrent touch events at the same time, is there an implicit preference in the system for some TX ID's to go through more often than the other's? Figure 10 confirms the fairness of RWB under four random backoff strategies. The figure also shows that shorter backoff times (i.e., more frequent transmission) also lead to loosing more messages in the communication.

The next question is the achievable temporal resolution for recognizing concurrent touch events. Since the random backoff time slows down the transmission pace compared to keeping the beaconing at constant, high frequency, some performance degradation compared to the single touch event recognition case is expected. Figure 11 shows the achievable overall system performance for four RWB configurations. The plot indicates that the strategy of using a random backoff between 3 ms and 15 ms generates higher throughput than other configurations – even though that version has the highest packet loss ratio.

6.4 End-to-end performance

We also present an end-to-end evaluation of a TangibleID application to understand the accuracy and latency from the users' perspective. In this evaluation we use an end-user application of TangibleID combined with the Unity AR framework. Although receiving and processing a BCC packet from the moment it was sent takes only approximately 2.68 ms (2.19 ms for baseband signaling, 0.49 ms for

---

[1] Over-the-air coupling can occur directly around BCC devices up to 10-20 cm's [52].

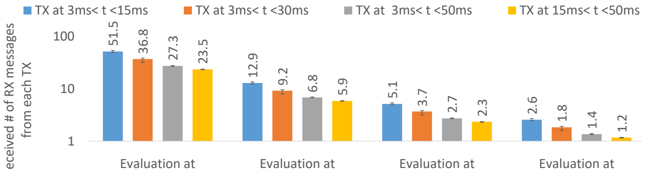


Figure 11: Temporal touch resolution under different multiple access strategies, for two TX and one RX setups, using a logarithmic scale. The results are presented in 4 different ways, where the size of the evaluation window varies between 50 ms and 1 sec.

Table 2: User evaluation of accurate object identification.

	Book	Map	Bottle	Painting
Touch events collected (#)	2592	2195	2666	2326
Correct ID received / touch	95.72%	94.72%	99.44%	99.40%
Avg touch duration (sec)	22.13	22.76	13.77	8.27

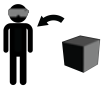
RX processing), the overhead of the bridging mechanism adds extra latency. By using a TouchCom – Overwatch – Unity application TCP pipeline, we measure an average of 123.01 ms delay (shortest 12.31 ms, longest 249.43 ms; from 84 measurements, with 5 outliers cut out) between the moment of touching a TouchCom tangible and that information showing up on a screen of the connected Unity application [2]. While *current research is mostly inconclusive about latency requirements* [43], and applications have different requirements, this performance is within the range of current touchscreen latencies [38].

To measure accuracy, 7 users were recruited to reach for 4 TangibleID objects. While holding an Android tablet, each was asked a total of 20 times to reach for a given object (all lying on a desk), in a randomly generated order. Each user was given an RX wristband, while each object was configured to constantly beacon its ID. The RX recorded each occurring touch event and constantly forwarded them (through Overwatch) to the Unity application. A total of 55.96 minutes of data was collected during this evaluation. Table 2 shows that for each tangible, it is possible to precisely detect and display the touched/picked-up object. The success rate reflects on how many occasions the incoming packets match the actual objects touched. Errors arise when the electric fields surrounding the objects are in too close proximity – as it seemed to be the case for the book and the map. In these cases, we can observe cross talk, which can be handled by the touch sensitivity threshold described earlier.

7 INTERACTION MODALITIES ENABLED

TangibleID provides a much more comprehensive capture of tangible interactions than was previously possible. In this section, we organize TangibleID's supported feature set in a unified manner. We describe various physical interaction configurations, and we discuss how they can be leveraged for an enriched analog/digital experience.

7.1 Base case



7.1.1 Context awareness. Users can extract identification of TangibleID-enabled objects while touching or holding them. This capability inherently gives context-awareness: we have precise information of what objects the user is interacting with. The end-user application (AR or otherwise), now with an understanding of the context, can better react to its participants.

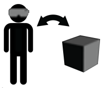
TangibleID's core feature is this contact-based *Context awareness.* While understanding which objects the user is picking up can be already interesting, this base case can be extended to a variety of configurations, as seen in Table 3.

Table 3: Comparison of TangibleID interaction modalities. In each case, the participants' user-/object-/location IDs are continuously known.

Category	Icon	Modality	Usage scenario
Base case		Context-awareness	augment based on what we are interacting with
Smart objects		Objects with memory / identity	tangibles as unique instances & reprogrammable artifacts
		Bridging objects	control augmented story with pairing tangibles
Reactive spaces		Persons at locations	augment based on who is where
		Objects at locations	augment based on what & where we are interacting with
Multiple users		Interacting with objects together	recognizing concurrent use of the same object
		Interpersonal interactions	interaction recognition between people for sharing information

7.2 Interactions with smart objects

Smart physical objects can own a digital identity, maintain state information and interaction history, and store digital media. Their data could be (read-write) accessed upon contact.



7.2.1 Objects with memory/identity. In contrast to object recognition through passive sensing, BCC supports arbitrary data exchange. This communication allows not just distinguishing instances of the same class of objects, but also supports having a readable state actively stored in the objects themselves. Leveraging this property can open up the possibility for *in the wild* experiences, where annotation data and other media can be extracted on-the-fly from the objects and displayed by a preferred method.

---

[2] To establish the ground truth on the moment of touch, for this measurement only, we also integrated a physical button onto the BCC electrodes – this way we could properly measure the whole end-to-end latency.

7.2.2 Bridging objects. Sometimes interactions with certain objects are not of particular interest or are too ambiguous. In these cases, the simultaneous presence of another object could resolve the recognition problem. TangibleID can recognize concurrent interaction with several objects and use this information at application level, for example, to influence its narrative (e.g., simultaneous interaction with action figures can change the storyline). From a technical point of view, this modality does not need explicit user instrumentation for the tangible interaction recognition, which can be an advantage if the user fluctuation is high.

7.3 Reactive spaces

Physical contact can be interpreted beyond using only hands for the interactions. A smart infrastructure (such as floor, desk) could sense and even identify users touching or walking on them. By adding this capability to an end-user application a truly immersive experience can be achieved, where – for example – the storyline of an entertainment application is influenced by the location.

7.3.1 Persons at locations. TangibleID supports integrating floor pieces that can pick up the user's ID to provide precise location information while a user walks on them. This capability can be leveraged in an application by triggering events when a user appears at specific locations. When the user ID is mapped to a user role, the application can trigger custom content for each user.

7.3.2 Objects at locations. Locations can be sensitive also to the presence of specific objects and vice versa. In this complex configuration, TangibleID can recognize if a user carries certain objects to certain location. Then the application or its narrative can adapt based on which objects appear where. Similarly to *Bridging objects*, the user's identity might be irrelevant here (hence user instrumentation on the BCC level can be skipped).

7.4 Multi-user scenarios

Interactions involving multiple people are prevalent in everyday life, and they are generally perceived as more engaging. However, there are only a few applications that enable direct interactions between users. The technical complexity behind tracking such interactions is usually too high to allow these interactions to naturally take place. Using TangibleID we can support a wider range of events.

7.4.1 Interacting with objects together. Certain experiences might only to be triggered when multiple users with different roles are present. To support this functionality, TangibleID objects can recognize user IDs during interaction, even if several people are interacting with them at once.

7.4.2 Interpersonal interactions. Understanding physical contact between users – such as a handshake, high-five, or simply holding hands – can unlock a series of functionalities. TangibleID does not only understand the timing of the contact, but knows the identity of the participants as well (e.g., for team selection or explicit consent to data exchange).

8 TANGIBLEID IN AUGMENTED REALITY

Augmented Reality (AR) is one of the primary applications that could benefit from TangibleID. Section 8.2 reports a full implementation of TangibleID with AR and its usage in the *Haunted Castle*; the next subsection discusses the AR specific aspects of the application development.

8.1 Vision, tracking, display

Since in most TangibleID experiences, users interact with physical objects, the ideal AR augmentation display leaves the users' hands free. A head-mounted display (HMD) or projected augmented reality satisfies this preference. However, for faster prototyping, our showcase was first implemented for an Android tablet (Samsung Galaxy Tab S3 9.7), which allows using wider viewing angles during development but limits physical interaction to one hand while the other hand holds the tablet. Unity [1] allowed us to easily port the application to a Microsoft HoloLens HMD afterwards. Object tracking was implemented using the Vuforia [2] AR toolkit, which also supports extended tracking, i.e., it maintains (for a while) the recognized AR markers' locations in the environment, even when the markers are not visible anymore due to the user simply looking in other directions.

User interaction with objects in the environment is the motivation to employ TangibleID in AR, but many interactions may have ramifications on the AR system that deals with the user's interactions. E.g., the AR markers on an object may temporarily be occluded. Extended tracking, unfortunately, cannot compensate for this temporary loss of line-of-sight to the marker, because its primary purpose is to compensate for the moving viewpoint, not to handle complete temporary occlusion. This issue can be addressed with a combined AR/BCC effort. Once the AR marker is recognized, the world coordinate of the marker should be saved. If the marker is not visible anymore, then we can check the touch channel. If BCC indicated we are in contact with the given object, and the IMU data of the object don't suggest movements, we can assume the object's position has not changed, its marker is just temporarily being occluded by the user's hand(s). Algorithm 1 summarizes the steps.

8.2 Showcase: The Haunted Castle

We re-imagine the idea of a well-known theme park attraction, by turning it into an immersive mixed reality experience. The *Haunted Castle* is a proof-of-concept TangibleID AR application, an immersive multi-player game, where users follow several scenes through a castle. The castle comes into life through the AR display (HMD or hand-held). Each user is equipped with a BCC device that holds its owner's unique ID, which maps to a specific role in the game. By understanding the players' physical interactions with the real world and incorporating them into the AR narrative, advanced storytelling possibilities open up. In the following, we describe how TangibleID enables novel interactions for the Haunted Castle, and we show how these interactions shape the AR experience.

8.2.1 Welcome to the Castle. In the first room, the players discover a painting with glowing eyes. The painting functions as *guest book*. When a player looks at the painting through the AR

Algorithm 1 Compensating for AR marker occlusion

```
$object$                ▷ IN/OUT: object data (recognized/location)
$marker$                ▷ IN: marker visibility/location in world coordinate
$contact$               ▷ IN: BCC value if object is currently being touched
$moving$                ▷ IN: accelerometer data of the object
function RecognizeObject($object, marker, contact, moving$)
    if $marker.visible = true$ then
        $object.recognized \leftarrow true$
        $object.location \leftarrow marker.location$
        $object.lastLocation \leftarrow marker.location$
    else if $contact = true \ \& \ moving = false \ \& \ object.lastLocation != invalidLocation$ then
        $object.recognized \leftarrow true$
        $object.location \leftarrow object.lastLocation$
    else
        $object.recognized \leftarrow false$
        $object.lastLocation \leftarrow invalidLocation$
    end if
end function
```

Figure 12: When users touch the painting (1), the current (trapped) inhabitants revealed: previous visitors (2), now joined by a new player (3).

display and touches it, earlier visitors who haven't escaped yet are revealed. Since the player is also a trapped visitor, their own picture is also depicted in the painting (Figure 12). The *Welcome to the Castle* scene relies on the property that smart objects can have their own identity and memory, which can be revealed upon interaction. When the painting is touched, it sends the list of past visitors and adds the current user's ID and picture once they are received through the touch (BCC) channel.

8.2.2 How to Deal with Ghosts. Objects picked up by users can influence the 3D content appearing in the environment. Here, when users hold the ghost book (BCC-tagged), it summons a ghost, which then appears in front of them. When users pick up a lantern (BCC-tagged), the ghost fades away (Figure 13). This example demonstrates a context-aware interaction with objects that are not necessarily in line-of-sight. TangibleID still recognizes when the interactions (touch, grasp, hold) happen and understands which object the user is interacting with (through BCC ID-exchange), allowing the AR narrative to adapt.

8.2.3 Capturing the Essence. By holding up the enchanted book and an enchanted bottle at the same time, the ghost can be captured. Both of these objects have AR markers so that they can be tracked and animated through the AR display (Figure 14). They also have a BCC tag. When the user holds up both objects at the same time, the human body inherently forms an electrical path between them, allowing objects to detect each other's presence – even without an explicit BCC device on the user. When the simultaneous presence is recognized, the AR animation can be triggered.

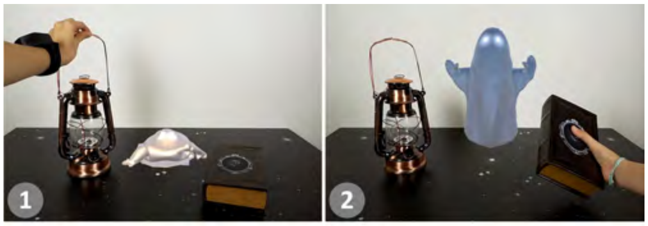


Figure 13: Picking up a lantern scares the ghost away (1). Picking up the ghost book summons the ghost back (2).

Figure 14: The ghost's essence is transferred from the book to the bottle when holding both objects.

8.2.4 Trap Door. This scene starts with a painting of a deer; the AR system recognizes the marker and renders a 3D deer head. In front of the painting, there is a hidden (virtual) trap door. The trap door is physically implemented as a BCC-floor tile. Depending on which user steps on the trap door, different animations are triggered. Users playing the `Ghost` role can *float* above the trap door, therefore they are not affected. Other players fall through the trap door (as displayed through the AR display) and end up in a basement with a barred window (Figure 15). This scene implements the idea of recognizing people at certain locations, and using this information to trigger different actions in the AR application for different people.

8.2.5 Alive Painting. The location also plays a key role in *Alive Painting*: whereas in *How to Deal with Ghosts* the 3D content follows the users floating through space, here we explore a static setup. A painting (used as AR marker) reacts to the user and their objects, but only if they are in close proximity. This time, the ghost book triggers Celia (the lady in the painting) being haunted, while the lantern relieves her from the ghost essence (Figure 16). To recognize that a user holding a (BCC tagged) book or with a (BCC tagged) lantern is nearby (and so to only trigger the AR animation in this situation), it is not enough to know that the user is interacting with objects. We also must know where that interaction takes place. This scene is an example of an objects-at-location interaction, with a BCC floor tile installed in front of the painting.

8.2.6 Hidden Map. The *Hidden Map* scene encourages users to interact with objects together. While a seemingly empty floor plan of the castle is found to be lying around, a palm print indicates that it has hidden content. However, for each user the map reveals different parts through the AR display. The whole map can

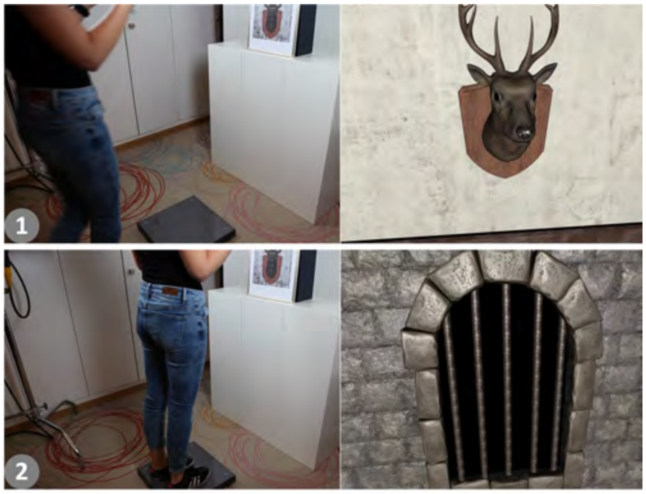


Figure 15: Depending on the user's role, the trap door activates when the user steps on it, and the user may fall into the dungeon.

Figure 16: Depending on what object the user brings close to the painting, the painting will have a different reaction.

be seen only when all users stack their hands together on the map. The map is a BCC tagged object that recognizes concurrent touch of different (BCC) users occurring at the same time (Figure 17). This scene shows an example of mediated interaction of multiple participants.

Figure 17: Map reveals information depending on who is interacting with it; multiple users can join forces to reveal hidden map.

Figure 18: Sharing secrets. When meeting Sir Charles, the player's AR system recognizes his name tag and shaking hands reveals more information.

8.2.7 Sharing Secrets. At last, a crew member impersonates Sir Charles, the mad scientist. He can be recognized through his name tag, which serves as an AR marker. Sir Charles can decide to reveal his secret formula on how to use ghost essence to gain immortality by offering a hand-shake to the players. By instrumenting Sir Charles with a BCC wristband, a direct interpersonal interaction can be implemented. Moreover, arbitrary digital messages can be sent through the handshake, such as the secret formula. Introducing the handshake into the AR application ensures that users have the opportunity to annotate information about themselves that requires explicit consent (i.e., handshake) before release (displaying to other users).

8.3 Preliminary user feedback

Eight users (4 female, 4 male) were recruited to experience the Haunted Castle. Each user anonymously answered a survey, using five-point Likert-type scale: 7 questions focused on the overall experience, and 7 evaluated each scene individually. Figure 19 shows that users found the TangibleID AR experience engaging and novel,

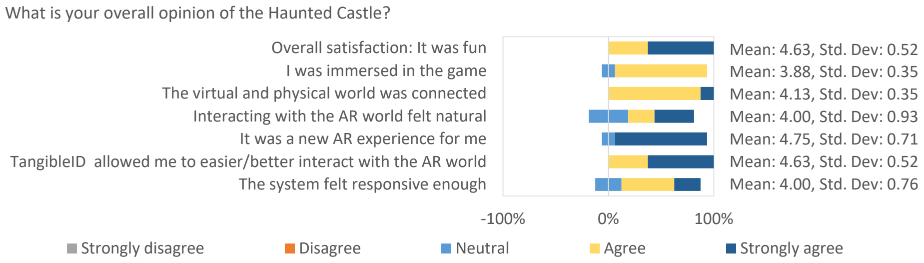


Figure 19: Overall user feedback on the Haunted Castle.

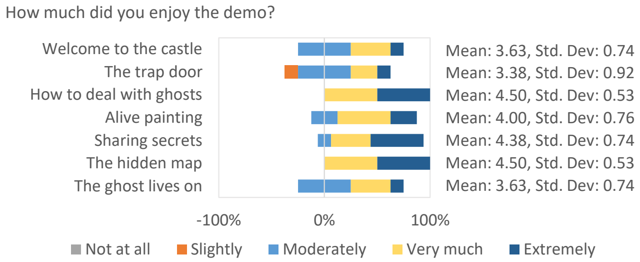


Figure 20: User feedback on the individual scenes.

and it allowed them to better (resp. more easily) interact with the AR world. Additional feedback also pointed out that interactions with people or with moving, animated objects are generally more interesting than static scenarios. During the study the users used a tablet as augmented display, for easier navigation, and because the HoloLens proved to have an uncomfortably narrow field of view.

9 DISCUSSION

The previous section demonstrates that TangibleID can successfully enhance mixed reality experiences but real-world limitations must be considered and improved upon on in the future. In the following we discuss some current limitations and then look towards potential future directions.

Accuracy and interference Identifying the interaction of closely placed BCC objects might suffer from unintentional coupling of the applied electric field between those objects, as seen in Table 2. This phenomenon of over-the-air coupling has been described earlier by BCC literature [53]. Compensating for this interference is non-trivial, however, choosing appropriate sensitivity thresholds might help.

Factors of end-to-end latency The end-to-end latency is determined by three factors: (1) BCC-level transmission and processing; (2) bridging mechanism (between BCC and end-user application); and (3) end-user application level processing. The third factor is often negligible, but the first and second factors might have a substantial impact on the observed latency. The bridging mechanism, in our implementation, for simplicity, is a star topology TCP/WiFi connection. This suboptimal solution explains why the best and worst cases differed significantly (12.31 ms vs 249.43 ms). However, these numbers also indicate that by optimizing the bridging mechanism (e.g., as per Section 5.2), much better performance can be achieved than the average 120 ms showed here. On the BCC-level, accuracy problems cause delayed detection, hence the transmission must be fairly fast while also guaranteeing accuracy. Accuracy problems usually arise from the unstable BCC channel. However, the interaction modalities as proposed in Section 7 rely on using the limbs, a choice that usually guarantees stable BCC signals [53].

Simultaneous usage Further evaluation is needed to understand how much the proposed system could scale with simultaneous users. Referring to the previous paragraphs on end-to-end latency: the bottleneck could arise from the bridging mechanism, e.g., from an overloaded WiFi network. In this case, once again, alternative bridging mechanisms or a network with dedicated channels may be used.

AR workflow and occlusion The integration of BCC itself does not interfere with the AR markers (or modifies the AR workflow), since the electrodes can be hidden right beneath the surface. However, the tangible interaction itself might occlude the AR markers – which is a generic problem in TAR applications. While Algorithm 1 provides a starting point on how to address this problem partially, a comprehensive and robust solution is still missing.

Alternative BCC devices While the paper discusses the design parameters and performance of TangibleID, the proposed architecture may apply to any system that wishes to use BCC technology for real-time capture of holistic tangible interactions. In particular, the following discussions stand independent from the BCC HW/SW artifacts:

- generic application workflow [Section 3];
- physical integration considerations, BCC electrode design [Section 4.1.2];
- software ecosystem design [Section 5], data bridging topologies for user-worn devices [Section 5.2];
- network protocol for BCC and its application in the real-time interaction capture [Section 6]. While the actual measurements of performance aspects might vary if we applied a different BCC HW/SW platform, the generic concepts remain the same: the concept of maximizing shortened message throughput [Section 6.1], the concept of translating data transmission to temporal resolution for touch

recognition [Section 6.2], the validation of the BCC channel properties on concurrent messaging and the associated concept of how to support concurrent touch events [Section 6.3], and the concept of how to measure end-to-end performance [Section 6.4];

- the enabled interaction modalities [Section 7].

Therefore, the difference between TangibleID and a TangibleID-like system (using alternative BCC devices) would be mainly in the end-to-end performance.

TangibleID beyond BCC The TangibleID prototype system's main purpose is to demonstrate the newly enabled interaction capabilities, focusing mostly on the BCC-enabled features. However, one could easily imagine higher utilization of sensor-fusion on board. Using IMUs or capacitive sensing, we could extend the interaction modalities to gesture recognition. This approach would allow to answer not only the *when*, *where*, and *with what do users interact* questions, but the *how do they interact* question as well.

10 CONCLUSION

TangibleID is a reference implementation of a novel architecture, consisting of physical prototypes and a software ecosystem, which opens novel opportunities for tangible interactions. This architecture allows applications (e.g., games based on AR) to incorporate real-time physical interaction with objects and people. TanbibleID uses Body Channel Communication to capture the precise timing of interaction events that occur in the physical world, independent of where they happen – inside or outside the field of view of any cameras. Because objects (or locations) can have identity and/or history, an application can distinguish between different objects even if they look (from the outside) identically, or can adjust the behavior based on past interaction histories. As a result, new interactive immersive experiences are possible as illustrated by the Haunted Castle showcase.

To support application development, not only must Body Channel Communication provide stable data transfers but it is important that the overhead of the software ecosystem stays low enough to allow (soft) real-time processing by applications. An extensive evaluation shows that TangibleID is capable of delivering robust, real-time performance suitable for interactive mobile applications.

TangibleID provides a novel way to close the gap between the physical and digital worlds. As application developers strive to realize new integrated experiences, frameworks such as TangibleID become increasingly important as a solid foundation to react to and manage real-life events, objects, and user interactions.

REFERENCES

[1] 2018. Unity. Website. https://unity3d.com/, accessed Sept 30, 2018.

[2] 2018. Vuforia - Augmented Reality. Website. https://www.vuforia.com/, accessed Sept 30, 2018.

[3] 2019. Qt – Cross-Platform Software Development. Website. https://www.qt.io/, accessed Mar 30, 2019.

[4] Matt Adcock, Matthew Hutchins, and Chris Gunn. 2003. Augmented Reality Haptics: Using ARToolKit for Display of Haptic Applications. In *Proceedings of 2nd IEEE International Augmented Reality Toolkit Workshop*. 1–2. https://doi.org/10.1109/ART.2003.1320415

[5] Teemu T. Ahmaniemi and Vuokko T. Lantz. 2009. Augmented Reality Target Finding Based on Tactile Cues. In *Proceedings of the 2009 International Conference on Multimodal Interfaces* (Cambridge, Massachusetts, USA) *(ICMI-MLMI '09)*. ACM, New York, NY, USA, 335–342. https://doi.org/10.1145/1647314.1647383

[6] Judith Amores, Xavier Benavides, and Pattie Maes. 2015. ShowMe: A Remote Collaboration System That Supports Immersive Gestural Communication. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI EA '15)*. ACM, New York, NY, USA, 1343–1348. https://doi.org/10.1145/2702613.2732927

[7] Ronald T. Azuma. 1997. A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments* 6, 4 (Aug. 1997), 355–385. https://doi.org/10.1162/pres.1997.6.4.355

[8] Gerald Bianchi, Benjamin Knoerlein, Gabor Szekely, and Matthias Harders. 2006. High Precision Augmented Reality Haptics. In *Proc. EuroHaptics*, Vol. 6. 169–178. https://www.vision.ee.ethz.ch/publications/papers/proceedings/eth_biwi_00455.pdf

[9] Mark Billinghurst, Adrian Clark, and Gun Lee. 2015. A Survey of Augmented Reality. *Found. Trends Hum.-Comput. Interact.* 8, 2-3 (March 2015), 73–272. https://doi.org/10.1561/1100000049

[10] Mark Billinghurst, Raphael Grasset, and Julian Looser. 2005. Designing Augmented Reality Interfaces. *SIGGRAPH Comput. Graph.* 39, 1 (Feb. 2005), 17–22. https://doi.org/10.1145/1057792.1057803

[11] Mark Billinghurst, Hirokazu Kato, and Ivan Poupyrev. 2001. The MagicBook – Moving Seamlessly between Reality and Virtuality. *IEEE Computer Graphics and Applications* 21, 3 (May 2001), 6–8. https://doi.org/10.1109/38.920621

[12] Doug A. Bowman, Ernst Kruijff, Joseph J. LaViola, and Ivan Poupyrev. 2004. *3D User Interfaces: Theory and Practice.* Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA. https://doi.org/10.1162/pres.2005.14.1.117

[13] Jongeun Cha, Ian Oakley, Junhun Lee, and Jeha Ryu. 2005. An AR System for Haptic Communication. In *Proceedings of the 2005 International Conference on Augmented Tele-existence* (Christchurch, New Zealand) *(ICAT '05)*. ACM, New York, NY, USA, 241–242. https://doi.org/10.1145/1152399.1152444

[14] Robert F. Cleveland Jr., David M. Sylvar, and Jerry L. Ulcek. 1997. Evaluating Compliance with FCC Guidelines for Human Exposure to Radiofrequency Electromagnetic Fields. https://transition.fcc.gov/bureaus/oet/info/documents/bulletins/oet65/oet65b.pdf

[15] Philip Cohen, David McGee, Sharon Oviatt, Lizhong Wu, Joshua Clow, Robert King, Simon Julier, and Lawrence Rosenblum. 1999. Multimodal Interaction for 2D and 3D Environments. *IEEE Comput. Graph. Appl.* 19, 4 (July 1999), 10–13. https://doi.org/10.1109/38.773958

[16] Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. 2007. MonoSLAM: Real-Time Single Camera SLAM. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 6 (June 2007), 1052–1067. https://doi.org/10.1109/TPAMI.2007.1049

[17] Junjun Fan, Xiangmin Fan, Feng Tian, Yang Li, Zitao Liu, Wei Sun, and Hongan Wang. 2018. What is That in Your Hand?: Recognizing Grasped Objects via Forearm Electromyography Sensing. *IMWUT* 2, 4 (2018), 161:1–161:24. https://doi.org/10.1145/3287039

[18] Tobias Grosse-Puppendahl, Sebastian Herber, Raphael Wimmer, Frank Englert, Sebastian Beck, Julian von Wilmsdorff, Reiner Wichert, and Arjan Kuijper. 2014. Capacitive Near-field Communication for Ubiquitous Interaction and Perception. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Seattle, Washington) *(UbiComp '14)*. ACM, New York, NY, USA, 231–242. https://doi.org/10.1145/2632048.2632053

[19] Tobias Grosse-Puppendahl, Christian Holz, Gabe Cohn, Raphael Wimmer, Oskar Bechtold, Steve Hodges, Matthew S. Reynolds, and Joshua R. Smith. 2017. Finding Common Ground: A Survey of Capacitive Sensing in Human-Computer Interaction. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. ACM, New York, NY, USA, 3293–3315. https://doi.org/10.1145/3025453.3025808

[20] Anuruddha Hettiarachchi and Daniel Wigdor. 2016. Annexing Reality: Enabling Opportunistic Use of Everyday Objects As Tangible Proxies in Augmented Reality. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 1957–1967. https://doi.org/10.1145/2858036.2858134

[21] Otmar Hilliges, David Kim, Shahram Izadi, Malte Weiss, and Andrew Wilson. 2012. HoloDesk: Direct 3D Interactions with a Situated See-through Display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. ACM, New York, NY, USA, 2421–2430. https://doi.org/10.1145/2207676.2208405

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

[23] Christian Holz and Marius Knaust. 2015. Biometric Touch Sensing: Seamlessly Augmenting Each Touch with Continuous Authentication. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Daegu, Kyungpook, Republic of Korea) *(UIST '15)*. ACM, New York, NY, USA, 303–312. https://doi.org/10.1145/2807442.2807458

[24] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms. In *Proceedings of the ACM SIGCHI Conference*

on Human Factors in Computing Systems (Atlanta, Georgia, USA) *(CHI '97)*. ACM, New York, NY, USA, 234–241. https://doi.org/10.1145/258549.258715

[25] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, and Andrew Fitzgibbon. 2011. KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. ACM, New York, NY, USA, 559–568. https://doi.org/10.1145/2047196.2047270

[26] Hirokazu Kato, Mark Billinghurst, Ivan Poupyrev, Kenji Imamoto, and Keihachiro Tachibana. 2000. Virtual Object Manipulation on a Table-Top AR Environment. In *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*. 111–119. https://doi.org/10.1109/ISAR.2000.880934

[27] Kiyoshi Kiyokawa, Haruo Takemura, and Naokazu Yokoya. 1999. A Collaboration Support Technique by Integrating a Shared Virtual Reality and a Shared Augmented Reality. In *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.99CH37028)*, Vol. 6. 48–53 vol.6. https://doi.org/10.1109/ICSMC.1999.816444

[28] Georg Klein and David Murray. 2007. Parallel Tracking and Mapping for Small AR Workspaces. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '07)*. IEEE Computer Society, Washington, DC, USA, 1–10. https://doi.org/10.1109/ISMAR.2007.4538852

[29] Benjamin Knoerlein, Gábor Székely, and Matthias Harders. 2007. Visuo-haptic Collaborative Augmented Reality Ping-pong. In *Proceedings of the International Conference on Advances in Computer Entertainment Technology* (Salzburg, Austria) *(ACE '07)*. ACM, New York, NY, USA, 91–94. https://doi.org/10.1145/1255047.1255065

[30] Gierad Laput, Robert Xiao, and Chris Harrison. 2016. ViBand: High-Fidelity Bio-Acoustic Sensing Using Commodity Smartwatch Accelerometers. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. ACM, New York, NY, USA, 321–333. https://doi.org/10.1145/2984511.2984582

[31] Gierad Laput, Chouchang Yang, Robert Xiao, Alanson Sample, and Chris Harrison. 2015. EM-Sense: Touch Recognition of Uninstrumented, Electrical and Electromechanical Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Daegu, Kyungpook, Republic of Korea) *(UIST '15)*. ACM, New York, NY, USA, 157–166. https://doi.org/10.1145/2807442.2807481

[32] Gun A. Lee, Gerard J. Kim, and Mark Billinghurst. 2007. Interaction Design for Tangible Augmented Reality Applications. In *Emerging Technologies of Augmented Reality: Interfaces and Design*. IGI Global, 261–282. https://doi.org/10.4018/978-1-59904-066-0.ch013

[33] Gun A. Lee, Claudia Nelles, Mark Billinghurst, and Gerard Jounghyun Kim. 2004. Immersive Authoring of Tangible Augmented Reality Applications. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '04)*. IEEE Computer Society, Washington, DC, USA, 172–181. https://doi.org/10.1109/ISMAR.2004.34

[34] Gun A. Lee, Theophilus Teo, Seungwon Kim, and Mark Billinghurst. 2017. Shared-sphere: MR Collaboration Through Shared Live Panorama. In *SIGGRAPH Asia 2017 Emerging Technologies* (Bangkok, Thailand) *(SA '17)*. ACM, New York, NY, USA, Article 12, 2 pages. https://doi.org/10.1145/3132818.3132827

[35] Minkyung Lee, Richard Green, and Mark Billinghurst. 2008. 3D Natural Hand Interaction for AR Applications. In *2008 23rd International Conference Image and Vision Computing New Zealand*. 1–6. https://doi.org/10.1109/IVCNZ.2008.4762125

[36] Woohun Lee and Jun Park. 2005. Augmented Foam: A Tangible Augmented Reality for Product Design. In *Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '05)*. IEEE Computer Society, Washington, DC, USA, 106–109. https://doi.org/10.1109/ISMAR.2005.16

[37] Hanchuan Li, Eric Whitmire, Alex Mariakakis, Victor Chan, Alanson P. Sample, and Shwetak N. Patel. 2019. IDCam: Precise Item Identification for AR Enhanced Object Interactions. In *IEEE International Conference on RFID, RFID 2019, Phoenix, AZ, USA, April 2-4, 2019*. 1–7. https://doi.org/10.1109/RFID.2019.8719279

[38] Albert Ng, Julian Lepinski, Daniel Wigdor, Steven Sanders, and Paul Dietz. 2012. Designing for Low-latency Direct-touch Input. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. ACM, New York, NY, USA, 453–464. https://doi.org/10.1145/2380116.2380174

[39] Alex Olwal, Hrvoje Benko, and Steven Feiner. 2003. SenseShapes: Using Statistical Geometry for Object Selection in a Multimodal Augmented Reality System. In *Proceedings of the 2Nd IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)*. IEEE Computer Society, Washington, DC, USA, 300–301. http://dl.acm.org/citation.cfm?id=946248.946836

[40] Thammathip Piumsomboon, David Altimira, Hyungon Kim, Adrian Clark, Gun Lee, and Mark Billinghurst. 2014. Grasp-Shell vs Gesture-Speech: A Comparison of Direct and Indirect Natural Interaction Techniques in Augmented Reality. In *2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 73–82. https://doi.org/10.1109/ISMAR.2014.6948411

[41] Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, and Andy Cockburn. 2013. User-defined Gestures for Augmented Reality. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems* (Paris, France) *(CHI EA '13)*. ACM, New York, NY, USA, 955–960. https://doi.org/10.1145/2468356.2468527

[42] Simon Prince, Adrian David Cheok, Farzam Farbiz, Todd Williamson, Nik Johnson, Mark Billinghurst, and Hirokazu Kato. 2002. 3D Live: Real Time Interaction for Mixed Reality. In *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work* (New Orleans, Louisiana, USA) *(CSCW '02)*. ACM, New York, NY, USA, 364–371. https://doi.org/10.1145/587078.587129

[43] Kjetil Raaen and Tor-Morten Grønli. 2014. Latency Thresholds for Usability in Games: A Survey. In *Norsk informatikkonferanse (NIK)*. https://ojs.bibsys.no/index.php/NIK/article/view/9

[44] Holger T. Regenbrecht, Michael Wagner, and Gregory Baratoff. 2002. MagicMeeting: A Collaborative Tangible Augmented Reality System. *Virtual Reality* 6, 3 (2002), 151–166. https://doi.org/10.1007/s100550200016

[45] Jun Rekimoto. 1996. Transvision: A Hand-Held Augmented Reality System for Collaborative Design. In *Proceeding of Virtual Systems and Multimedia*, Vol. 96. 18–20. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.50.9615

[46] Jun Rekimoto and Katashi Nagao. 1995. The World Through the Computer: Computer Augmented Interaction with Real World Environments. In *Proceedings of the 8th Annual ACM Symposium on User Interface and Software Technology* (Pittsburgh, Pennsylvania, USA) *(UIST '95)*. ACM, New York, NY, USA, 29–36. https://doi.org/10.1145/215585.215639

[47] Munehiko Sato, Ivan Poupyrev, and Chris Harrison. 2012. Touché: Enhancing Touch Interaction on Humans, Screens, Liquids, and Everyday Objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. ACM, New York, NY, USA, 483–492. https://doi.org/10.1145/2207676.2207743

[48] Orit Shaer and Eva Hornecker. 2010. Tangible User Interfaces: Past, Present, and Future Directions. *Found. Trends Hum.-Comput. Interact.* 3, 1-2 (Jan. 2010), 1–137. https://doi.org/10.1561/1100000026

[49] James Vallino and Christopher Brown. 1999. Haptics in Augmented Reality. In *Proceedings of the IEEE International Conference on Multimedia Computing and Systems - Volume 2 (ICMCS '99)*. IEEE Computer Society, Washington, DC, USA, 9195–. https://doi.org/10.1109/MMCS.1999.779146

[50] Virag Varga. 2019. *Reinventing Touch with Body Channel Communication – System Design from Electric Fields to Mixed Reality.* Ph.D. Dissertation. ETH Zurich.

[51] Virag Varga, Gergely Vakulya, Alanson Sample, and Thomas R. Gross. 2017. Playful Interactions with Body Channel Communication: Conquer It!. In *Adjunct Publication of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. ACM, New York, NY, USA, 81–82. https://doi.org/10.1145/3131785.3131798

[52] Virag Varga, Gergely Vakulya, Alanson Sample, and Thomas R. Gross. 2018. Enabling Interactive Infrastructure with Body Channel Communication. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 169 (Jan. 2018), 29 pages. https://doi.org/10.1145/3161180

[53] Virag Varga, Marc Wyss, Gergely Vakulya, Alanson Sample, and Thomas R. Gross. 2018. Designing Groundless Body Channel Communication Systems: Performance and Implications. In *The 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. ACM, New York, NY, USA, 683–695. https://doi.org/10.1145/3242587.3242622

[54] Edward Jay Wang, Jake Garrison, Eric Whitmire, Mayank Goel, and Shwetak Patel. 2017. Carpacio: Repurposing Capacitive Sensors to Distinguish Driver and Passenger Touches on In-Vehicle Screens. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. ACM, New York, NY, USA, 49–55. https://doi.org/10.1145/3126594.3126623

[55] Shuangquan Wang, Jie Yang, Ningjiang Chen, Xin Chen, and Qinfeng Zhang. 2005. Human activity recognition with user-free accelerometers in the sensor networks. In *2005 International Conference on Neural Networks and Brain*, Vol. 2. 1212–1217. https://doi.org/10.1109/ICNNB.2005.1614831

[56] Zhengyou Zhang. 2012. Microsoft Kinect Sensor and Its Effect. *IEEE MultiMedia* 19, 2 (Feb 2012), 4–10. https://doi.org/10.1109/MMUL.2012.24

[57] Thomas G. Zimmerman. 1996. Personal Area Networks: Near-field Intrabody Communication. *IBM Syst. J.* 35, 3-4 (Sept. 1996), 609–617. https://doi.org/10.1147/sj.353.0609