Using Mobile Phones to Spontaneously Authenticate and Interact with Multi-Touch Surfaces

Johannes Schöning
Institute for Geoinformatics
University of Münster
Robert-Koch-Str. 26-28
48149 Münster, Germany
j.schoening@uni-muenster.de

Michael Rohs
Deutsche Telekom Laboratories
TU Berlin
Ernst-Reuter-Platz 7
10587 Berlin, Germany
michael.rohs@telekom.de

Antonio Krüger
Institute for Geoinformatics
University of Münster
Robert-Koch-Str. 26-28
48149 Münster, Germany
antonio.krueger@uni-muenster.de

ABSTRACT

The development of FTIR (Frustrated Total Internal Reflection) technology has enabled the construction of large-scale, low-cost, multi-touch displays. These displays—capable of sensing fingers, hands, and whole arms—have great potential for exploring complex data in a natural manner and easily scale in size and the number of simultaneous users. In this context, access and security problems arise if a larger team operates the surface with different access rights. The team members might have different levels of authority or specific roles, which determines what functions they are allowed to access via the multi-touch surface. In this paper we present first concepts and strategies to use a mobile phone to spontaneously authenticate and interact with sub-regions of a large-scale multi-touch wall.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Input devices and strategies, interaction styles.

Keywords

Multi-touch interaction, frustrated total internal reflection, large displays, mobile devices, input strategies, authentication, emergency scenario, CSCW.

1. INTRODUCTION & MOTIVATION

Multi-touch interaction with computationally enhanced surfaces has received considerable attention in the last few years. The rediscovery of the FTIR principle, which allows for building such surfaces at low cost, has pushed the development of new large-scale multi-touch applications fast forward. These walls are well suited for multi-user collaboration with large data sets, such as geographical or time-stamped data. In scenarios with large surfaces (i.e. more than 2 meters) and large groups of users (i.e., more than two) controlling access to content and functionality made available through the multi-touch surface is often an important requirement. However, although FTIR allows identifying a large number of contact points on the wall, it does not discriminate between different users. This makes it difficult to control who is issuing a command. This can lead to severe security problems if the multi-touch wall is used for triggering real-world events, as is the case in control room scenarios. For example, in an emergency response to a flooding event (cf. [9]), where a team of experts needs to coordinate mobile forces on the ground (e.g., fire brigades) and monitor data on a geographical representation (e.g., flood level and degree of pollution of air and water), not all users should be able to manipulate all data presented on the multi-touch wall. Depending on the particular policy, only the commander of the fire brigade forces might be allowed to send a mobile unit to a new target (e.g., by pointing to the unit and the new destination). Authentication concepts known from desktop computing are not well suited for these settings, since they usually grant access to an application or the whole computer, rather than to a local area of the screen.

In this paper we are addressing the problem that in some collaborative work situations the group of users of a multi-touch wall varies greatly in competence, hierarchical level, and decision-making authority, demanding a dedicated authentication and access mechanism for small regions of a multi-touch surface. We present a first solution for how to authenticate a user who wants to interact with a sub-region of a multi-touch wall. We present novel concepts that enrich the interaction with multi-touch surfaces by using a personal mobile device to spontaneously authenticate and interact with the multi-touch wall.

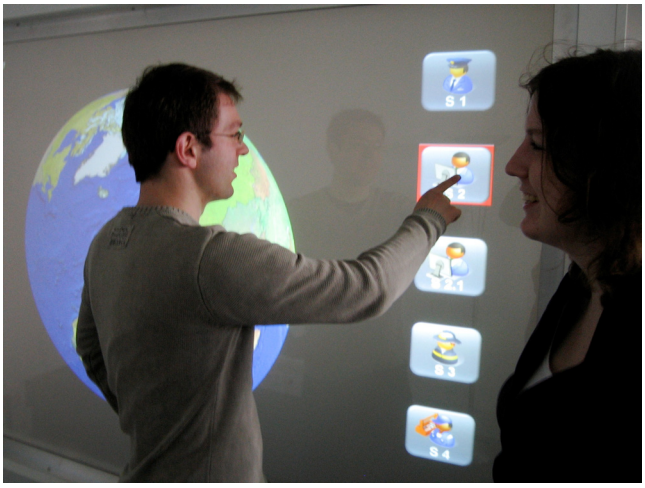

Figure 1: Multi-user interaction with a multi-touch wall in an emergency scenario without dedicated access control: The user is selecting an authentication level by pressing a button representing a certain role.

The paper is structured as follows: First we briefly give an overview of related work. In Section 3 we introduce an authentication concept using the flashlight and Bluetooth unit of a mobile device as response channels. Due to the fact that we did not yet run user tests on the interaction we discuss some possible variants of the basic concept, which we intend to evaluate in the future. We also present more general ideas for how to enrich the functionalities of a large scale multi-touch wall using mobile devices in an emergency setting. In Section 3 we briefly summarize the state of implementation. In the last section we present our conclusion and ideas for future work.

2. RELATED WORK

Collaborative visualization and decision-making tools for crisis response has been a classical field of the Digital Cartography, Visualization and GIS communities. In addition, other disciplines, such as the HCI and Ubiquitous Computing communities, have tried to tackle various aspects of this problem. Most of the existing work focuses on large format map applications that support decision-making, for example, in an emergency operation center (EOC). McEachren [5] et al. provide a good overview of these large format map applications that support collaborative visualization and decision-making. The GIS wallboard [3] is a conceptual example of an electronic white board envisioned to support sketch-based gestures to interact with geospatial data. Sharma et al. [8] concentrate on multi-modal interaction (speech and gesture) with a large dynamic map display and evaluated that system in a crisis response scenario with real users. All this work concentrates on supporting decision-making and group collaboration in an EOC, but does not concentrate on the problem of multi-user interaction with different levels of authority. An interesting alternative to classical input devices, like mice and keyboards, especially in emergency scenarios is multi-touch technology, which allows multi-finger and bimanual operation [1], because in such scenarios users have to make large-scale decisions very quickly and definitely. Several hardware solutions exist that allow the realization of multi-touch input on surfaces of different sizes. Buxton[1] gives a thorough overview of current technologies as well as the history of multi-touch surfaces.

Jeff Han presented the original FTIR multi-touch sensing work in February 2006 at the Technology Entertainment Design (TED) Conference [4]. This technology has the advantages in that it can be constructed from readily available components, is cheap and can be scaled without problems to a large scale multi-touch wall. Using this technology, multi-touch surfaces can be easily integrated into EOC where users often interact with geospatial information. However, FTIR surfaces just detect touch events and do not provide the identity of the users, per se. If multi-touch applications need to distinguish between different users, the *Diamond Touch* concept from MERL [2] could be used, with the drawback that the users either need to be wired or stay in specially prepared locations. Because an EOC is a very dynamic work setting and users have to be flexible and switch between different work stations, such a technology is not useful for an emergency scenario. We have determined that the benefits of using FTIR far outweigh the disadvantage that it does not identify users.

Mayrhofer et al. [6] present a method for establishing and securing spontaneous interactions on the basis of spatial references which are obtained by accurate sensing of relative device positions. In their work they implemented an interlocked protocol using radio frequency messages and ultrasonic pulses for verifying that two devices share a secret.

3. USER IDENTIFICATION & AUTHENTICATION

As already motivated in the introduction, collaborative work at a multi-touch surface often involves users with different roles, competencies, and scopes of expertise. In an emergency response scenario, for example, a media contact person may be allowed to visualize statistical data on the wall to get an up-to-date picture of the situation, while only the officer-in-charge may command emergency troops at the real emergency site. It would thus increase safety and security if the system could distinguish between users or if individual input events could be authenticated. This would also help in a later analysis of the events that took place, since critical operations could be attributed to individual users.

Even in such a scenario we would like to retain the direct-touch interaction scheme of FTIR multi-touch surfaces as much as possible. We assume that most interactions are allowed for every user and that only a small subset of interactions are critical, e.g., because they trigger external real-world events such as sending troops to a specific position. It therefore seems to be acceptable if these critical operations require a slightly higher interaction effort than the other operations.

The minimum requirement to support the above scenario is to identify the user who generates the critical input event. The system could then check whether the identified user is

[1] `http://www.billbuxton.com/multitouchOverview.html` (2008)

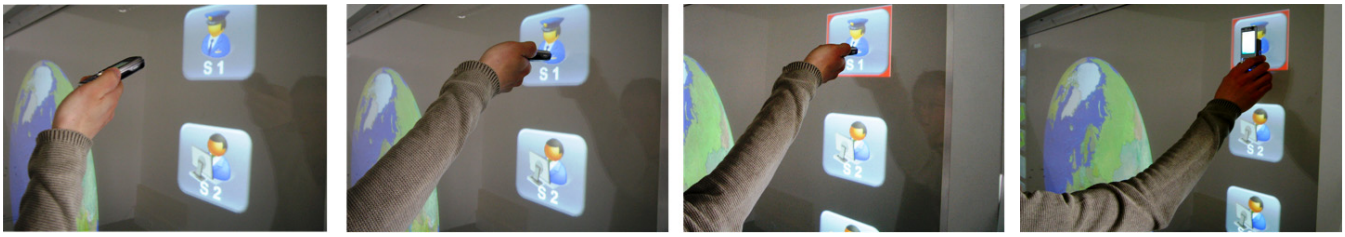

Figure 2: Interaction scheme to authenticate with a specific user role on an FTIR multi-touch surface: (i) The user touches the wall with the phone. (ii) The mobile phone flash light sends a light flash (or a camera flash) to indicate the region the user wants to interact with and at the same time initializes the authentication process. (iii+iv) The user can interact in his/her assigned role with the wall and do critical actions.

authorized to trigger the associated action. A better solution would be to also cryptographically authenticate the user attempting the input action instead of mere identification. Of course, it would be best to continuously authenticate each individual contact point, e.g., each contact point during a dragging operation. However, this is not possible given bare finger input and current FTIR technology. It is also not necessary for enabling scenarios like the one outlined above. A solution in which a user "logs in" to a small region in order to gain exclusive access to the region until the user releases that region again does not seem to be adequate, because we assume that, in general, quick access to all parts of the multi-touch surface is required.

We therefore propose to identify—and if possible also authenticate—users in the case of critical operations by using a mobile device as a mediator. We assume that the device contains a flash light and Bluetooth connectivity, and is able to detect touch events with an integrated microphone or accelerometer. We further assume that the FTIR system has a second camera that detects light flashes in the visible range. The basic identification scheme (without cryptographic authentication) works as follows:

1. The user touches a region of the wall with the phone.
2. The phone detects the touch event with its built-in accelerometer or microphone and generates a light flash. Simultaneously it sends the user ID via Bluetooth. (Optionally, microphones can be installed at the multi-touch surface as proposed in [7] to determine the position of touch event on the surface.)
3. The surface detects the light flash at a certain position and receives the user ID via Bluetooth. The light flash can be distinguished from finger touch events, because it produces a bright light strobe in the visible range, whereas finger touch events are detectable mainly in the infrared range.
4. The surface either detects the light flash first or receives the user ID via Bluetooth first. Both events have to be received within a short time window Δt. If either one is missing or if they are more than Δt apart, the protocol is aborted. If more than one flash event and one ID event are detected during a time window extending from Δt before the first event and Δt after the second event, this is considered as a collision.
5. If a collision was detected the server asks one of the devices that have sent an ID to repeat the procedure. Here also random backoff procedures could be used to resolve the collision, in which the device waits a random amount of time before a retransmission is attempted (c.f. Ethernet media access).
6. If a unique association of position and user ID is found the server looks up the authorization data for the object at the respective position and checks whether the user is allowed to perform the action. If so, a positive response is sent via Bluetooth and the action is executed. In addition, visible feedback on the region is given to indicate success or failure.

The above algorithm uniquely identifies input events on individual regions, even with multiple simultaneous users generating finger input events and multiple users generating phone touch events. If a user touches some other object this will generate only a Bluetooth ID event, but no flash event will be detected by the surface, so the algorithm will abort or a collision with another user will happen. The algorithm is guaranteed to uniquely associate user identities to regions if both events are generated and sensed within Δt.

A shortcoming of this algorithm is that it is not cryptographically secure. An attacker could forge a user ID and thus execute unauthorized critical operations on behalf of another user. We identified the following requirements for an algorithm that authenticates input on a sub-region of the multi-touch wall to support the above scenario:

- The main goal is to ensure that critical operations are only executed by authorized users. The authentication scheme thus has to prove the identity as well as the input position of the user who attempts the operation.
- The system should log all critical interactions for later analysis and documentation. Ideally, the system should also ensure non-repudiation of critical interactions. It should be possible to reconstruct who was responsible for which interaction.
- The system should allow for easy and spontaneous authentication without requiring too much effort and without interfering with other simultaneous users who perform non-critical operations.

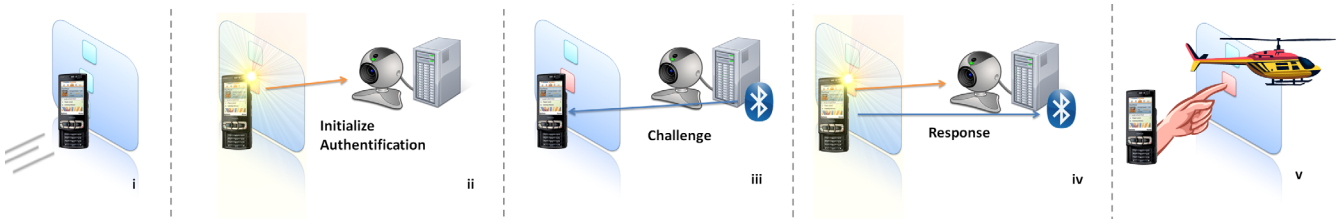

Figure 3: General interaction scheme to identify a user with a certain area on an FTIR multi-touch surface: (i) The user touches the wall with the phone. (ii) The mobile phone flash light sends a light flash (or a camera flash) to indicate the region the user wants to interact with and at the same time initializes the authentication process. (iii+iv) The user is identified can interact in his/her assigned role with the wall and do critical actions. The more detail scheme is described in the body of that paper.

With Bluetooth we have a high bandwidth connection but we cannot determine the position on the multi-touch surface where the user actually touched the surface. With the flash light we have a very low bandwidth data channel and way to detect the input position. We assume that the multi-touch surface server and all mobile devices that are allowed to interact with the surface have a pair of cryptographic keys—a public key, a private key, and a corresponding certificate.

We propose the following preliminary authentication scheme. In order to prevent forging, the user ID is signed with the private key of the mobile device before sending it to the server. To prevent replay attacks a timestamp and a sequence number are included in the authentication request. The authentication protocol proceeds as follows:

1. The user touches a region of the wall with the phone.
2. The phone detects the touch event with its built-in accelerometer or microphone and generates a light flash. Simultaneously it sends the message m via Bluetooth:

   $$m = enc(R', pubKey_{server})$$

   with

   $$R' = (R, sign(hash(R), privKey_{device}))$$

   $$R = (opcode, userID, time, seq.nr., rand.delay)$$

   $$opcode = inputrequest$$

   We assume that only the device knows $privKey_{device}$ and thus only it is able to generate a valid "input request" message.
3. The surface detects the light flash at a certain position and receives m via Bluetooth. If the content of m cannot be verified it is discarded. Verification includes the signature, the timestamp, and the sequence number for that device.
4. As above, if more than one flash event and one ID event are detected during a time window extending from Δt before the first event and Δt after the second event, this is considered as a collision.
5. As above, if a collision was detected the process is repeated.
6. As above, authorization is performed and feedback is given accordingly.

We assume that a valid signature of the message sent via Bluetooth can only be generated by the device containing the private key. Therefore the server can be sure that a successfully verified ID stems from an authentic input request. If an attacker produces or replays an input request, verification will fail at the server. However, an attacker can produce flash events. If we assume that the authentic device produces a flash event as well, the attacker can only produce a collision.

A problem occurs, if a device generates an input request, but the corresponding light flash is not detected by the surface. This could happen if a touch event is triggered while not facing the surface. In this case the light flash would never reach the surface and an attacker could produce a light flash on some random display region.

To solve this problem, a second light flash could be produced after a random delay whose duration is sent in m (see step 2 above). The attacker would then have to guess the right delay and produce the second flash at exactly the right moment. If the server detects a flash before the indicated delay, the procedure is aborted. The security of this approach depends on the accuracy with which the camera can detect the light flashes. In the current setup, the camera runs at 30 Hz, which severly limits the bandwidth of the visual channel. An obvious way to get a higher bandwidth is to increase the frame rate of the camera. We are also working on other solutions. One idea is to introduce a light back channel. A challenge could be sent by projecting a pattern on the surface next to the detected light spot. The camera of a mobile device is normally located next to the light flash and could detect the challenge and send it back to the server (signed and encrypted). This approach has the advantages that the back channel via the mobile device camera has a higher bandwidth and we can be sure that the user is actually interacting with the right sub-regions of a large-scale multi-touch wall.

For the implementation we use a Nokia 5500 with a built-in flash light and the Nokia N95 using its built-in camera flash. A camera image (recorded by a DragonFly camera with an infrared filter) of the raw camera image and the N95 touching the multi-touch surface can be seen in Figure 4.

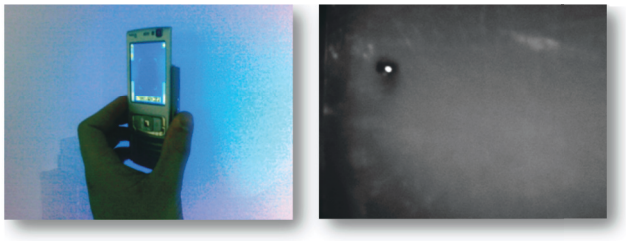

Figure 4: User is touching the multi-touch wall with a mobile device. Raw camera image of the phone flash using a DragonFly camera with an infrared filter.

4. CONCLUSIONS AND FUTURE WORK

We addressed the problem of spontaneous authentication of individual input actions in the context of large-scale multi-touch FTIR surfaces. We described an access mechanism for small sub-regions of the surface that is capable of authenticating multiple simultaneous users. Users have to touch the wall with their personal mobile device for spontaneous authentication and interaction. We still have to do user tests on the usability and general acceptability of the proposed scheme. We intend to do a formal security analysis of the method and to evaluate it with real users in an emergency operation center. As future work, we plan to add additional functionality to our prototype. As an example, while the user touches the surface, the front camera can take a photo of the user and we can verify if the right user acts with the mobile device. Other functionalities beyond the authentication problem can be easily added. For example, in our emergency response scenario, a secured voice call connection could be easily established by touching the icon of a first responder troop on the surface. We also experiment with other output modalities like the display light or, if available, the IrDA port.

5. ACKNOWLEDGMENTS

The authors would like to thank Chris Kray for fruitful discussions during CHI 2008.

6. REFERENCES

[1] W. Buxton and B. Myers. A Study in Two-handed Input. *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 321–326, 1986.

[2] P. Dietz and D. Leigh. DiamondTouch: A Multi-user Touch Technology. *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pages 219–226, 2001.

[3] J. Florence, K. Hornsby, and M. Egenhofer. The GIS wallboard: Interactions with Spatial Information on large-scale Displays. *International Symposium on Spatial Data Handling*, 7:449–463, 1996.

[4] J. Y. Han. Low-cost Multi-touch Sensing through Frustrated Total Internal Reflection. In *UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology*, pages 115–118, New York, NY, USA, 2005. ACM.

[5] A. Maceachren and I. Brewer. Developing a Conceptual Framework for Visually-enabled Geocollaboration. *International Journal of Geographical Information Science*, 18(1):1–34, 2004.

[6] R. Mayrhofer, H. Gellersen, and M. Hazas. An Authentication Protocol using Ultrasonic Ranging. Technical Report COMP-002-2006, Lancaster University, October 2006.

[7] J. Paradiso and C. Leo. Tracking and Characterizing Knocks Atop Large Interactive Displays. *Sensor Review*, 25(2):134–143, 2005.

[8] R. Sharma, I. Poddar, E. Ozyildiz, S. Kettebekov, H. Kim, and T. Huang. Toward Interpretation of Natural Speech/Gesture: Spatial Planning on a Virtual Map. *Proceedings of ARL Advanced Displays Annual Symposium*, pages 35–9, 1999.

[9] SoKNOS Project. `www.soknos.de`, 2008.