Acoustic Barcodes: Passive, Durable and Inexpensive Notched Identification Tags

Chris Harrison **Robert Xiao** **Scott E. Hudson**

Human-Computer Interaction Institute and Heinz College Center for the Future of Work
Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213
{chris.harrison, brx, scott.hudson}@cs.cmu.edu

ABSTRACT

We present acoustic barcodes, structured patterns of physical notches that, when swiped with e.g., a fingernail, produce a complex sound that can be resolved to a binary ID. A single, inexpensive contact microphone attached to a surface or object is used to capture the waveform. We present our method for decoding sounds into IDs, which handles variations in swipe velocity and other factors. Acoustic barcodes could be used for information retrieval or to triggering interactive functions. They are passive, durable and inexpensive to produce. Further, they can be applied to a wide range of materials and objects, including plastic, wood, glass and stone. We conclude with several example applications that highlight the utility of our approach, and a user study that explores its feasibility.

ACM Classification: H.5.2 [Information interfaces and presentation]: User Interfaces - Graphical user interfaces; Input devices and strategies.

General terms: Design, Human Factors

Keywords: Sound, vibration, microphones, identification, ID, tags, markers, classification, location, interaction techniques, ubiquitous and pervasive computing.

INTRODUCTION

Our world is increasingly "tagged" to facilitate information retrieval and to trigger interactive functions. For example, UPC optical barcodes are ubiquitous on consumer goods and QR Codes are being used in smartphone applications. We propose an alternative tagging scheme based on acoustics (i.e., mechanical vibration). Specifically, we use patterned, physical tags composed of a series of parallel grooves (Figure 1). These can be etched, embossed or cut into a variety of surfaces and objects. When a ridged object such as a fingernail is dragged over these notches, they produce bursts of sound that encode a unique ID (Figure 2).

This approach has unique qualities. Foremost, users can trigger acoustic barcodes with their (un-instrumented) fingers. It is also possible to use rings, pens, dry erase markers, mobile phones, keys and many other implements. Moreover, tags are passive, durable, and inexpensive to mass-produce. We used a $6 microphone for sensing, which can monitor roughly $10m^2$ of surface area, for example, a large whiteboard or table. Surfaces and objects can be augmented and easily retrofitted with acoustic barcodes. On some surfaces, tags can be made invisible. Overall, they can be smaller and subtler than visual markers (e.g., fiducial markers [10]). Finally, acoustic barcodes can be incorporated in a wide variety of materials.

UIST '12, October 7–10, 2012, Cambridge, Massachusetts, USA.

RELATED WORK

Researchers have explored many approaches for encoding unique identifiers. For example, visual schemes are popular, including 1D barcodes [11], 2D barcodes [19], and fiducial markers [10,17]. Identity can also be time-encoded using infrared light [18]. RFID tags use radio waves (electromagnetic radiation) for identification. Data can also be magnetically encoded, for example, the black strips on credit cards.

Most related to our technique are acoustic or tactile coding schemes. Braille [2] is a human-readable tactile encoding. The Edison Phonograph [4] is an analog system for recording and playing back sound by means of grooves cut in tinfoil. Moving from analog to digital: the Cricket System [15] uses coded ultrasound pulses to locate and identify users in an instrumented room. A listening device is attached to the user's hand and calculates its position in the room by measuring its distance to a set of fixed, coded emitters.

Another approach is acoustic-feature-driven classification. Hambone [3] is a wrist-worn acoustic sensor that detects movement of the arms via bone conduction, and performs gesture recognition by classifying the various sounds produced. Similarly, Skinput [9] uses an acoustic-sensing armband to localize finger taps on the skin (pre-learned locations). TapSense [8] and Sonically Enhanced Touch [12]

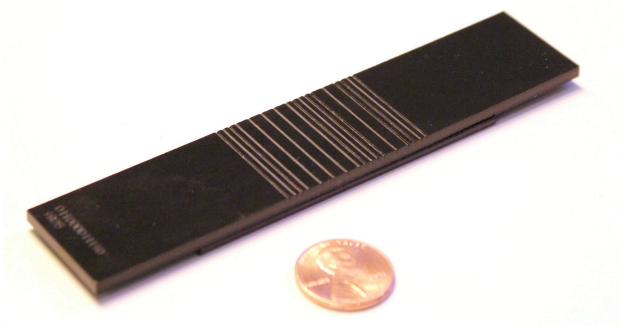

Figure 1. An acoustic barcode patterned into an acrylic tag.

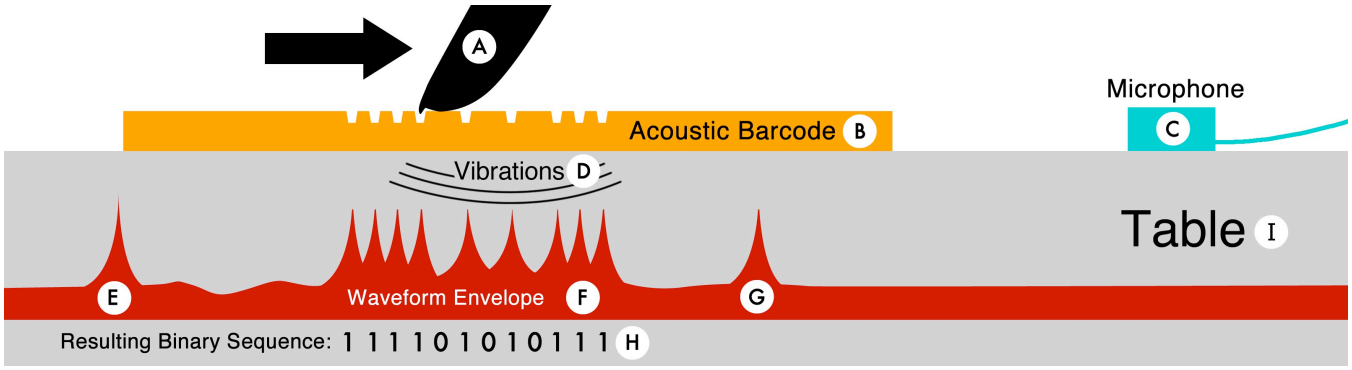

Figure 2. In this example setup, we have an acoustic barcode (B) situated on a table (I), to which a microphone (C) is attached. When a fingernail (A) runs over the barcode, a series of mechanical vibrations result (D), which propagate through the table and are captured by the microphone. The first sound is the initial impact of the finger (E). As the nail passes over the notches in the acoustic barcode, a series of sharp bursts of sound are produced (F). Finally, the finger lifts off or falls of the end of the barcode (G). The resulting waveform is processed, resulting in a decoded binary sequence (H).

use a microphone attached to an interactive surface (e.g., touchscreen) in order to differentiation among taps from different objects and parts of the finger, for example, the knuckle or nail. Stane [13] is a small palm-shaped device with an internal microphone and a multitude of engineered textures on its surface; the device classifies sounds produced by rubbing different areas, which can trigger interactive events. Scratch Input [7] uses a microphone situated on a surface to classify gestures performed on the surface, such as taps, swipes and shapes. Finally, Tapper [14] uses an array of microphones placed on a large sheet of glass to localize taps through time-of-flight correlation.

BARCODE CONSTRUCTION AND ENCODING

An acoustic barcode consists of a series of notches cut into a smooth host material. We use notches that are 0.25 to 0.5mm thick and 0.1 to 0.3mm deep. These are spaced 1.6 or 3.2mm apart. We recommended that notches should be at least 7mm wide to accommodate a fingernail or stylus.

In our encoding scheme, the spacing between notches is variable – the relative lengths of these gaps encode the data payload of the barcode. Specifically, the notches are separated by a small integer multiple of a unit gap distance, either 1.6mm or 3.2mm. Barcodes begin and end with a guard sequence consisting of three grooves separated by unit gaps. This guard sequence serves both to delineate the barcode and to provide a reference for the gap distance.

Following the guard, the payload is encoded. We provide for two different binary encoding schemes: a fixed-notch-count scheme and a fixed-physical-length scheme. Both encodings encode a single bit-sequence of fixed length. The bit sequence may be preprocessed with an error-correction scheme (such as a Hamming code [6] or Reed-Solomon code [16]) to improve the robustness of recognition.

In the fixed-notch-count scheme, 0's are represented as notches followed by a gap of one unit length. Conversely, 1's are notches followed by a gap of two unit lengths. This encoding uses a fixed number of notches, but produces a code that is of variable physical length. Alternatively, the fixed-physical-length scheme encodes each 1 as a notch and each 0 as a blank (i.e., no notch). Every bit is separated by exactly one unit gap. To avoid excessively long runs without notches, we constrain the space of permissible bit sequences to exclude sequences with two consecutive 0 bits. This better assures that a "clock signal" can be recovered, which is used to resolve the binary sequence. Note that this encoding has a fixed physical length, but uses a variable number of notches.

IMPLEMENTATION

We use an inexpensive off-the-shelf piezo contact microphone. An adhesive backing allowed the microphone to be attached to a wide variety of surfaces, such as windows, whiteboards and wooden tables. In general, contact microphones are sensitive only to mechanical vibrations on surfaces to which they are attached, and are robust to ambient noise. The signal is amplified and fed to a conventional laptop computer. Our approach, however, is computationally simple, opening up the possibility of implementation on an inexpensive embedded platform.

Capturing and Filtering Audio

We sample audio at 96kHz to reduce the likelihood of missing the brief transients produced by swiped notches. The audio signal is preprocessed with a high-pass filter at 4KHz to remove human speech and background hum (Figure 3, blue). The system listens for the onset of a swipe by maintaining a moving average of the amplitudes of the last 40ms of audio (4096 samples). Recording initiates if the average exceeds -40dBFS. Once recording begins, the system maintains a moving average of the last 250ms of audio, and stops the recording if this average goes below -40dBFS, or the recording exceeds 1000ms in length.

We then filter the recorded audio to make peak detection more robust. First, we transform the recorded audio by taking the absolute amplitude of each sample. Then, we compute an exponentially-weighted moving average over the recording to remove stray peaks and smooth the waveform slightly. Next, we take the sampled first derivative over the

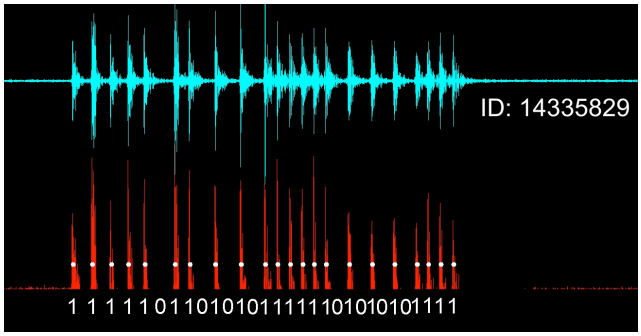

Figure 3. Blue: Example audio signal with 4kHz high pass filter applied. Red: final waveform used for peak detection. White dots indicate segmented peaks - the resolved 24-bit binary sequence is shown below (numerical ID on right).

smoothed recording, which accentuates peaks and subdues constant regions. Finally, we subtract a heavily smoothed copy of the recording from the derivative, which has the effect of removing regions of near-constant amplitude and secondary "echoes" from primary peaks.

Peak Detection

The filtered audio is fed to a peak detection routine (Figure 3, red). Iteratively, the peak detector takes the strongest amplitude value not within 300 samples (3ms) of another identified peak. By enforcing a minimum peak spacing, we avoid extracting echoes as independent peaks. The algorithm stops extracting peaks once the highest peak level is less than 3dB below the *K*th largest peak, where *K* is a lower bound on the number of notches in the barcode. This approach ensures that the algorithm handles variable-notch-count barcodes, while correctly identifying peaks that are stronger or weaker than the average.

Barcode Recovery

A list of time-stamped peaks is fed to a barcode decoder, which recovers the relative gap lengths between peaks. Peaks followed by gaps of 30ms or more are removed. These are assumed to be the product of errant peaks, like those produced when hitting the barcode prior to swiping it (Figure 2, G and E). Next, the peak list is filtered to remove gaps that are much shorter than the median gap length, since they likely represent echoes.

Since the first three notches of the barcode are separated by unit gaps, the decoder assumes that the first gap is the unit length. For each subsequent gap, the system computes the nearest integer multiple of the unit length, and emits that multiplier as the detected relative gap length. When a user swipes an acoustic barcode, it is rarely at a uniform velocity. Thus, reliable decoding must compensate for timing skew (see Figure 3; compare gap length at the start and end of the sequence). To compensate, we calculate the unit length implied by each gap (the gap length divided by the unit multiple). This is then averaged with the previous unit estimates, allowing the value to drift as decoding proceeds.

Materials

Part of our explorations included testing different materials for feasibility (Figure 4). Using a laser cutter, we have been

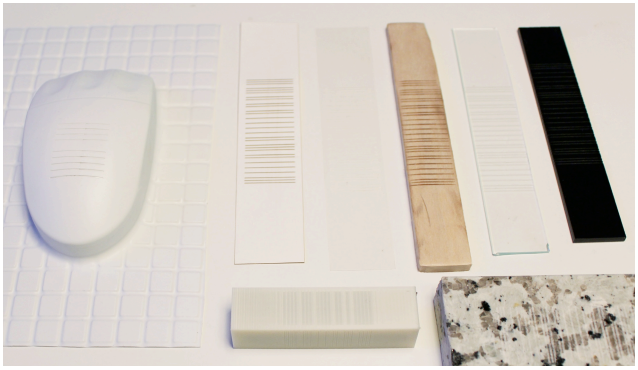

Figure 4. Acoustic barcodes can be incorporated into a variety of materials. Clockwise from top left: Polystyrene (vacuum-formed), paper, transparency, wood, glass, acrylic, granite, and a 3D printed object.

able to etch acoustic barcodes into acrylic, glass, wood, granite, paper, and transparencies (type 1 polyester). The latter two materials are flexible and can be applied to surfaces and objects like stickers. We also successfully laser etched acoustic barcodes into vacuum-formed polystyrene. Acoustic barcodes can also be directly incorporated into 3D printed objects. It is important that a high-density, gloss setting is used. Indeed, an important commonality between these successful materials is a smooth finish. A rough finish will produce excessive noise during swipes, which will obscure sounds produced by notches. There was minimal wear in the materials tested, suggesting acoustic barcodes are durable. Finally it seems likely that metal could be embossed with acoustic barcodes, though we were unable to test this.

EXAMPLE APPLICATIONS

We devised four example applications, which highlight different aspects, capabilities, and configurations of acoustic barcodes. Please also see the Video Figure.

Acoustic barcodes can be easily patterned into objects (e.g., part of the injection mold for plastic toys). To demonstrate this, we modified a wooden toy ship. Using a pen instrumented with a microphone, users can swipe over different parts of the ship (Figure 5A), which causes the name of the location to be read out loud, for example "port side" or "engine room". This could be easily extended to human models for learning about anatomy, labels on engine parts, and books with audio content.

We also created an acoustic-barcode enhanced storefront (Figure 5B). We fashioned clear acrylic tags and attached them to a large glass window (though ideally the window itself would be patterned). A microphone is attached to the inside of the glass (i.e., all "active" components are safe inside). Passersby can swipe the tags with their nails; the system reads back a description of the item and the price.

Most mobile devices have an internal microphone, which could be used to read acoustic barcodes in the environment - like an acoustic equivalent of QR Codes. As a proof of concept, we attached a microphone to an iPod Touch and created several office widgets. These allow a user to quickly

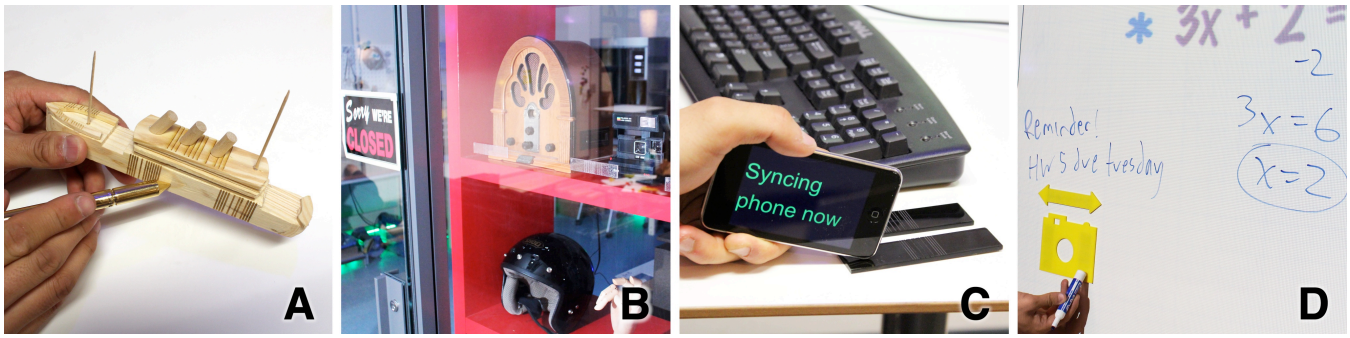

Figure 5. We created four example applications to highlight different uses of acoustic barcodes.

set the state of a device and the office, for example to "sync phone now" (Figure 5C) and "forward calls to desk."

Finally, we created an example suite of magnetically-backed widgets that can be stuck to whiteboards (Figure 6). The surface of these widgets is patterned with different acoustic barcodes. For example, the four-way arrow has different IDs on each direction and the circle has IDs around its periphery. Teachers, for example, could use these to create class-specific, ad hoc interfaces. Figure 5D shows a left/right arrow used to control a slide deck, and a camera widget used to trigger a photo of the whiteboard to provide students a copy of the non-digital work.

EVALUATION

To evaluate the effectiveness of acoustic barcodes we performed a small user study. Our goal in this study was two-fold. First, we wished to measure the raw accuracy of the recognition system, to determine if the approach was feasible. Second, we wished to empirically determine how much error correction might be required to provide a usable level of accuracy. For reference, 29x29 QR Codes have a raw payload of 841 bits, of which 281 bits are format information, fixed patterns, and other header data, 208 bits are usable data, and 352 bits are used for error correction.

We recruited 7 participants (four female) with a mean age of 26. Participants were paid $10 for their involvement. Before the experiment, participants were given a brief introduction to the acoustic barcodes technique.

Participants were asked to swipe six different types of barcodes with three devices: a fingernail, a dry erase marker, and a mobile phone. The microphone was affixed to the whiteboard in the fingernail and marker conditions and to the phone in the phone condition. The barcodes varied in bit count (6, 12, and 24 bit codes) and unit gap length (1.6mm and 3.2mm). All barcodes used in the study were encoded using the fixed-physical-length encoding scheme to avoid confounding effects from varying swipe distances.

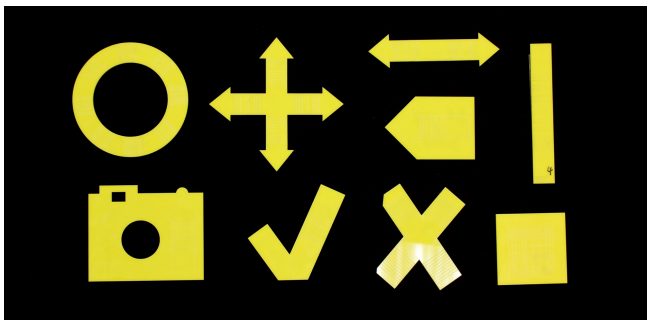

Figure 6. Widgets patterned with acoustic barcodes.

For each barcode type and device, participants performed a block of 15 trials, swiping 5 different bars 3 times each, for a total of 270 trials over the course of the experiment. Prior to starting each block, participants trained on two random unused bars to familiarize themselves with the swiping technique. The order of blocks was randomized to compensate for order effects. Participants completed all blocks for a single device before moving onto the next device. Participant swipes were processed in real time using the system described above.

RESULTS AND DISCUSSION

We calculated the Hamming distance [6] between each decoded bit sequence and the corresponding expected sequence, including the three guard bits at each end of the sequence. This yields the number of differing bits between the expected and generated sequences.

First, we looked at "raw" zero-error accuracy, that is, sequences that match exactly (Figure 7, left). The phone performed the best of our three input modalities. Across all bit counts and gap lengths, the phone condition correctly resolved 87.4% of the acoustic barcodes without any errors. Fingernail performs second best at 77.9%. Surprisingly, the dry erase marker performed worst overall, achieving an average zero-error recognition accuracy of 66.4%.

Importantly, any real world application would employ error correction for improved robustness. One issue that must be considered when adding error correction is the ratio of data bits to error-correction bits. Adding more corrective bits increases robustness, but decreases the number of available payload bits. Returning to our earlier example of a 29x29 QR Code, over 60% of its available data bits are devoted to error correction, resulting in just a third of its bits being used for data transmission. It should be noted, however, that QR Codes have significantly more data payload owing to their 2D nature (1D data density is comparable).

Correcting a single bit error in a 6-bit code can be done by using a truncated (7,4) Hamming code [6] to produce a barcode with 3 data bits and 3 check bits. Correcting two bit

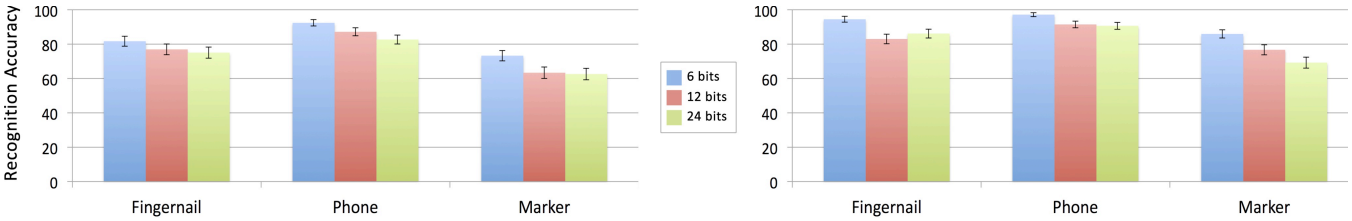

Figure 7. Left: Acoustic barcode recognition rates with perfect recognition (i.e., zero bit errors). Right: Recognition rates when error correction is included in the payload.

errors in a 12-bit code can be done using a truncated (15,7) BCH code [1], giving 4 data bits and 8 check bits. Finally, correcting three bit errors in a 24-bit code can be achieved by using an extended Golay code [5], which yields 12 data bits and 12 check bits, offering 4096 unique acoustic barcodes. Figure 7 (right) illustrates the recognition rates of our acoustic barcodes using these error-correcting schemes. The phone, fingernail and marker conditions achieve average accuracies of 93.1%, 87.4% and 77.3% respectively.

Increasing the size of the barcodes beyond 24 bits will increase the number of expected errors. However, our study suggests that the number of errors will grow linearly with the number of bits (around 10% of bits), which can be corrected with block codes such as Reed-Solomon [16]. For example, using a (63,30) BCH code, a 63-bit sequence, tolerant of up to 6 error bits, provides 30 data bits, offering one billion unique acoustic barcodes. Moreover, we believe that the unit gap length could be further reduced, enabling higher bit densities (there was no significant effect between the two gap lengths we evaluated). With a unit gap length of 1mm (down from 1.6mm), 60 bits can be encoded in roughly 6cm (2.4 inches).

The results in Figure 7 reflect a combination of system and user error. Anecdotally, approximately 5% of swipes were performed incorrectly by participants (e.g., started too close to the guard bits, fell off the side of the tag, ended swipe in middle of tag, used the pad of their finger instead of the nail). When this occurred, the signal was garbled or missing, precluding recognition. Over greater distances, it is harder to maintain a smooth motion, which reduced the accuracy of longer tags. Such user error is inherent in any system. No doubt every person has experienced having to swipe his or her subway or ATM card more than once. Even with recognition error rate of 10%, two swipes will suffice 99% of the time (99.9% for three, and so on).

CONCLUSION

We have described our work on acoustic barcodes, an identifying tag that uses notches to produce sound when dragged across. These could be patterned into a wide variety of materials and incorporated into everyday surfaces and objects. We used our system to prototype four application scenarios. We concluded with an evaluation of our approach, along with a discussion of how the data payload could be coupled with existing error correcting schemes.

ACKNOWLEDGMENTS

Funding for this work was provided in part by grants from NSERC, The Center for the Future of Work, Heinz College, CMU, and also a Microsoft Ph.D. Fellowship.

REFERENCES

1. Bose, R.C., Ray-Chaudhuri, D.K. On A Class of Error Correcting Binary Group Codes. *Information and Control*, 3(1), March 1960. 68–79.
2. Braille, Louis. (1829). Method of Writing Words, Music, and Plain Songs by Means of Dots, for Use by the Blind and Arranged for Them.
3. Deyle, T., Palinko, S., Poole, E.S., and Starner, T. Hambone: A Bio-Acoustic Gesture Interface. In *Proc. ISWC '07*. 3-10.
4. Edison, T. "Improvement in Phonograph or Speaking Machines." USPTO Patent #200521. 1878.
5. Golay, M. J. E. Notes on Digital Coding. In *Proc. IRE*, vol. 37, 1949. 657.
6. Hamming, R.W. Error Detecting and Error Correcting Codes. *Bell System Technical Journal*, vol. 29, Apr. 1950. 147-160.
7. Harrison, C. and Hudson, S.E. Scratch Input: Creating Large, Inexpensive, Unpowered and Mobile finger Input Surfaces. *In Proc. UIST '08*. 205-208
8. Harrison, C., Schwarz, J., and Hudson, S.E. TapSense: enhancing finger interaction on touch surfaces. In *Proc. UIST '11*. 627-636.
9. Harrison, C., Tan, D., and Morris, D. Skinput: appropriating the body as an input surface. In *Proc. CHI '10*. 453-462.
10. Kaltenbrunner M., and Bencina, R. reacTIVision: a computer-vision framework for table-based tangible interaction. In *Proc. TEI '07*. 69-74.
11. Lange, B.M., Jones, M.A., Meyers, J.L. Insight lab: an immersive team environment linking paper, displays, and data. In *Proc. CHI '98*. 550-557.
12. Lopes, P., Jota, R., and Jorge, J.A. Augmenting touch interaction through acoustic sensing. In *Proc. ITS '11*. 53-56.
13. Murray-Smith, R., Williamson, J., Hughes, S., and Quaade, T. Stane: synthesized surfaces for tactile input. In *Proc. CHI '08*. 1299-1302.
14. Paradiso, J.A. and Leo, C.K. Tracking and Characterizing Knocks Atop Large Interactive Displays. *Sensor Review*. 25(2), 2005, 134-143.
15. Priyantha, N.B. Chakraborty, A., and Balakrishnan, H. The Cricket location-support system. *Proc. MobiCom '00*. 32-43.
16. Reed, I.S. and Solomon, G. Polynomial Codes over Certain Finite Fields. *J. SIAM*, 8(2), 1960. 300–304.
17. Rekimoto J., and Ayatsuka, Y. CyberCode: designing augmented reality environments with visual tags. In *Proc. DARE '00*. 1-10.
18. Roth, V., Schmidt, P., and Guldenring, B. The IR ring: authenticating users' touches on a multi-touch display. In *Proc. UIST '10*. 259-262.
19. QR Code. Automatic identification and data capture techniques. Bar code symbology. BS ISO/IEC 18004:2006.