Humantenna: Using the Body as an Antenna for Real-Time Whole-Body Interaction

Gabe Cohn[1,2], Dan Morris[1], Shwetak N. Patel[1,2,3], Desney S. Tan[1]

[1]Microsoft Research
Redmond, WA (USA)
{dan, desney}@microsoft.com

[2]Electrical Eng., [3]Computer Science & Eng.
UbiComp Lab, DUB Group, Univ. of Washington
Seattle, WA (USA)
{gabecohn, shwetak}@uw.edu

ABSTRACT

Computer vision and inertial measurement have made it possible for people to interact with computers using whole-body gestures. Although there has been rapid growth in the uses and applications of these systems, their ubiquity has been limited by the high cost of heavily instrumenting either the environment or the user. In this paper, we use the human body as an antenna for sensing whole-body gestures. Such an approach requires no instrumentation to the environment, and only minimal instrumentation to the user, and thus enables truly mobile applications. We show robust gesture recognition with an average accuracy of 93% across 12 whole-body gestures, and promising results for robust location classification within a building. In addition, we demonstrate a real-time interactive system which allows a user to interact with a computer using whole-body gestures.

Author Keywords

Whole-body gestures, body as antenna, electrical noise

ACM Classification Keywords

H.5.2 Information interfaces and presentation: User Interfaces - Input devices and strategies.

INTRODUCTION

There is growing interest in new human-computer interfaces that go beyond the traditional keyboard or mouse and that are not mediated by special devices. The Xbox Kinect is an example of a commercially available input device that enables free-space whole-body gesture interaction using depth sensing and computer vision [16]. The commercial success of this device and the success of computer vision in general has stimulated the imaginations of consumers and researchers alike, and has led to rapid growth in explorations that leverage this new capability (e.g. see [15]).

However, the burden of installation and cost make vision-based sensing devices hard to deploy broadly, for example, through an entire home or building. Recognizing this limitation, researchers have explored sensors that leverage characteristics of the human body for sensing. Harrison et al. demonstrated the use of bio-acoustic sensors to determine the location of taps on the body [7]. Saponas et al. use electrical recordings of forearm muscles to sense muscle activity and infer finger gestures, which could be extended to other parts of the body [12]. Rekimoto presented a system that used capacitive sensing built into a wristwatch to sense finger gestures [11]. Additionally, more traditional approaches have used inertial sensors on the body for tracking whole-body gestures [8]. However, these on-body input systems are limited to gestures performed by the parts of the body on which the sensors are placed, and are not particularly effective for recognizing whole-body gestures.

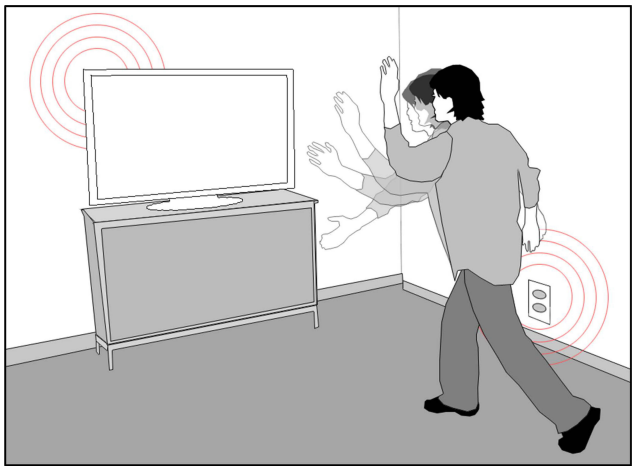

Figure 1: Prior work has shown that we can use the body as an antenna to turn uninstrumented walls into interactive surfaces [3]. We extend this technique and show that we can sense free-space whole-body gestures in real time.

In this work, we present Humantenna, an on-body sensing system that recognizes whole-body gestures. Humantenna works by using the human body as an antenna that receives existing electromagnetic (EM) noise from the power lines and electronic devices in a building (see Figure 1). Specifically, we use changes in the observed signal that occur as the body moves to different poses. In addition to demonstrating the ability to recognize various whole-body gestures in real-time, we also show robust classification of the person's location within the building among a small set of trained locations. This approach to sensing mobile whole-body interaction requires no instrumentation to the environment, and only minimal instrumentation to the user.

CHI'12, May 5–10, 2012, Austin, Texas, USA.

This paper provides significant advancements over previous work using the body as an antenna [3] with the following specific contributions:

1) We describe equipment and a set of offline processing techniques required for using the human body as an antenna for recognizing *whole-body* gestures.
2) We present a set of experiments conducted with 8 people in 8 homes showing the robustness of classifying a variety of free-space whole-body gestures.
3) We describe how we extend our offline techniques to perform automatic segmentation and real-time classification of whole-body gestures, and demonstrate how this can be used in interactive user interfaces.

BACKGROUND AND RELATED WORK

Whole-Body Gesture Sensing

Traditional methods of whole-body gesture recognition have largely used computer vision or inertial sensors. Computer vision provides a measurement method that does not require the user to wear any additional devices (e.g., [10]). The Xbox Kinect, which uses a hybrid RGB and depth sensing approach to extract body poses from the scene for detection of body gestures, has gained recent popularity [16]. In fact, this is part of a larger trend of emerging depth cameras and pixel-mixed devices (PMDs) that helps to alleviate some of the challenging problems encountered in traditional computer-vision, such as body segmentation.

Even with these new devices, vision-based approaches are generally limited in their field-of-view, are often sensitive to lighting, and suffer from occlusion problems. While researchers have looked at using thermal imaging coupled with RGB cameras to address challenges with lighting [9], occlusion is hard to overcome. Computer-vision techniques are also hard to scale throughout larger spaces such as homes, since this requires installing multiple cameras or sensors throughout the environment. A slightly different approach is used for applications that require extremely precise tracking: motion capture [14]. These systems typically use reflective markers placed at various locations on the body, but also require significant instrumentation to both the environment and the body.

Inertial sensing approaches locate all of the sensing on the body, which removes the requirement to instrument the environment [8]. However, sensing whole-body gestures may become cumbersome for users since it requires placing multiple sensors all over the body. Researchers have taken coarser approaches for detecting a subset of gestures using only a single device that a person is likely to already carry, such as a mobile phone placed in a pocket or held in a hand [1, 13]. However, depending on the location of the sensor, only gestures involving part of the body may be detectable.

In our work, we attempt to remove the requirement of instrumentation to the environment, and enable whole-body gesture sensing anywhere in the home or building, without any occlusion problems, using only a single sensor that can theoretically be placed anywhere on the body.

Interactive Surfaces Using the Body as an Antenna

Recently, Cohn et al. have proposed using the human body as an antenna for sensing touch gestures on walls and appliances [3]. The human body acts as an antenna and receives electromagnetic (EM) noise already present in the environment. Noise sources include the AC power signal, which is at either 50 or 60 Hz depending on the country, and higher frequency signals from appliances and electronic devices such as switch mode power supplies and dimmers [4]. Much of this noise is radiating from power lines, and can be picked up by the body [2, 5]. Using this noise as signal, they demonstrated that it is possible to infer touch gestures on walls and appliances throughout the home [3].

While impressive, this previous work was primarily limited to touch gestures. In addition, all touch segmentation was done manually, and with a very large feature-set (i.e., 1002 features every 82 ms), which made for excellent results when processed offline, but would be impractical for real-time use. In this paper, we significantly extend that work, leveraging the body as antenna and demonstrating our ability to sense and recognize free-space whole-body gestures. We also demonstrate a method for automatic segmentation and real-time classification of gestures, which can be integrated into interactive applications.

THE HUMANTENNA DEVICE

In this work, we extended the apparatus used in [3] to measure and digitize the voltages picked up by the human body. In reflecting on their work, the authors cautioned that data collection hardware, especially the laptop that they had in their backpacked setup, may have been a noise source that was inadvertently creating useful signal for classification. In addition, the size of the laptop could have created significant ground coupling to the body, which might have produced exaggerated performance over what would be expected on a small mobile device.

To alleviate this risk, and to move towards a form factor that is more representative of a mobile device that a user may carry, we miniaturized the setup and performed storage and computation off-board, rather than using a laptop carried by the user. Pragmatically, removing the laptop made the apparatus that our participants carried much smaller and lighter, and thus led to less fatigue in our experiments.

Specifically, we use a National Instruments WLS-9206 isolated wireless data acquisition unit, which is not as small as a mobile phone, but measures only 9.5 x 18.2 x 3.7 cm and is therefore significantly smaller than the laptop used in previous work. The unit takes voltage samples at 250 kS/s (kilo-samples per second) and digitizes them at 16-bit resolution. The data acquisition unit was modified to be powered using a 1000 mAh 3-cell Lithium ion polymer battery, which can power the system for several hours of continuous data collection. We biased the voltage on the human body contact point to a local ground signal on the data acquisition

unit through a 10 MΩ resistor in order to remove most of the DC offset of the single-ended voltage measurement.

We wirelessly transmitted the data captured and digitized by the acquisition unit over an IEEE 802.11g (Wi-Fi) communications channel to a computer placed elsewhere in the environment. This computer stored and processed the wireless data stream. In subsequent sections, we discuss offline as well as online processing schemes that we have applied to this data in various settings.

As with previous work, we make electrical contact to the neck of the user using a standard grounding strap, typically worn around the wrist when working with sensitive electronics. A small wire was used to connect the contact pad to the data collection equipment located in a backpack worn by the participant. While probably not the final attachment point in an ideal form factor, the neck is a convenient place for testing because it does not move much as a person gestures with their arms and legs, and it is near the data collection equipment located in a backpack.

EXPERIMENT 1: SENSING WHOLE-BODY GESTURES

We developed a system that could take data from our Humantenna device and (1) segment when a person is moving in free-space, and (2) classify the whole-body gesture the person is performing. To test this system, we performed an experiment in which participants in different homes conducted a number of gestures while wearing a Humantenna device. This experiment provides a proof-of-concept for our technique of sensing whole-body gestures. In this experiment, data was processed offline (i.e., after the data collection phase); however, later in the paper we describe a real-time implementation which extends on these techniques.

We conducted the experiment in 8 homes in the Pacific Northwest region of the United States, selected to represent a variety of constructions. These homes were all single-family residences built between 1964 and 2003 (μ=1984). They ranged in size between 195 and 288 square meters (μ=247), and had between 2 and 3 floors, some of them basements. For a single home, experiments were done in a single visit, although not all homes were tested on the same day. We used a different participant in each of these 8 homes. These 8 participants (2 female) were between 24 and 62 years old (μ=35), weighed between 50 and 79 kg (μ=68), and were between 150 and 180 cm tall (μ=169 cm).

Experimental Procedure

We collected data at two different locations in each home. One location was in the kitchen of the home, and the other location was in the family room. We chose these rooms because they existed in every home and offered very different environments for gesturing. The kitchen was typically a more confined area with many large appliances around the participant. On the other hand, the family room was typically a large open space with few electronics aside from a television and entertainment equipment, which were all turned off during data collection. In all homes, both locations were on the same floor and within sight of each other.

We chose a spot within each room where there was enough space to conduct all of the gestures in the experiment. To help participants remember the spot, we placed a small square of tape on the floor where the user was to stand to perform the gestures. The tape served as a guide to the general area of where to stand, but we did not instruct the users to stand in the same way in each repetition, nor did we instruct the users how they should use the tape to help them stand in the same location.

To minimize the number of variables that changed during the experimental session, we turned off the heating and air conditioning system, which can cause large changes in the electromagnetic noise in the home. However, we left smaller electronic devices, many of which also continually change their state, on during the experiment. We observed that many appliances, including refrigerators, hot water heaters, and computers changed their state during data collection. We turned on all lights in each of the rooms where data was collected, and did not manually change the state of lights or appliances once the experiment started.

Before beginning the experiment, we asked participants to empty their pockets of electronics and conductive materials and to remove their shoes to keep data collection as consistent as possible. Exploring any effects of these variables remains future work.

During the data collection phase, software running on the remote computer issued commands to the participant to guide them through the study. The software instructed participants to move to a given location and instructed them to perform each gesture. Once the researcher who was administering the study observed that the participant was in the start position for the given gesture and was not moving, they pressed a key on the computer, which issued a beep after a 0.5 second delay. This beep signaled the participant to perform the gesture, and then hold their body in the end position until another beep occurred. This second beep was produced 0.75 seconds after the researcher observing the study indicated that the participant had reached the end point of the gesture and was no longer moving. The delay between the researcher's key presses and the beeps were added to ensure that the user was not still moving when the beeps occurred. The timestamps of the beeps are used to frame the gesture, and help with the offline segmentation.

Participants first performed 12 gestures in a specified order at one of the locations, which we call a *run*. They repeated 4 runs at each of 2 locations, which we call a *session*. Participants performed 10 sessions, for a total of 960 whole-body gestures per participant.

We chose a gesture set which tests a variety of different types of movement. The gestures varied in duration between 3.3 and 7.6 s, with a median of 4.8 s. The 12 gestures used in the study are shown in Figure 2, and included:

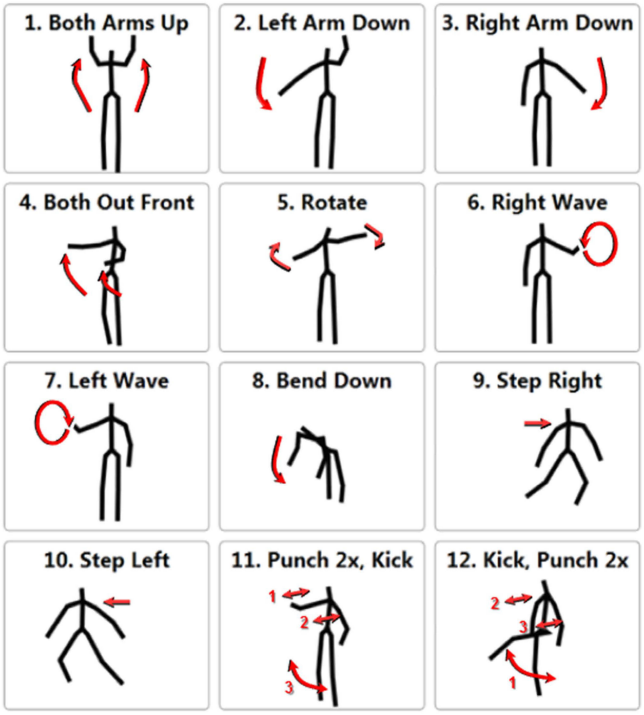

Figure 2: Stick figures depicting mirror images of the 12 gestures performed by all participants in experiment 1.

1. Starting from a rest position standing up with both arms at their sides, participant moved both arms simultaneously up until both were above the head
2. With both arms still above the head, participant brings the left arm back down to their side
3. With the right arm still above the head, participant brings it back down to their side
4. With both arms starting at the sides, participant moves them simultaneously outward in front until they are parallel to the ground
5. Participant twists torso so they are facing to the right with their left arm out in front of their body and their right arm out behind their body. Then, they rotate counter-clockwise 180 degrees until the right arm is directly in front of the body and the left arm directly behind
6. Starting with both hands in front of the chest, participant performs a counter-clockwise circular wave with the right hand
7. Starting with both hands in front of the chest, participant performs a clockwise circular wave with the left hand
8. Standing with both arms down at their sides, participant bends down as if they are touching their toes
9. From a standing position, participant takes one step to the right, leaving their left foot planted
10. From a standing position, participant takes one step to the left, leaving their right foot planted
11. Participant performs a right punch, followed by a left punch, and then a kick with the right foot
12. Participant performs a kick with the right foot, followed by a right punch, and a then a left punch

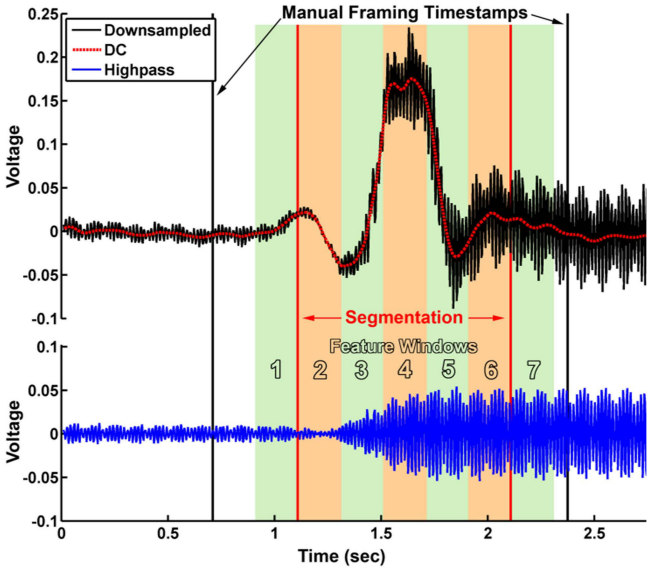

Figure 3: Down-sampled data (solid black), DC waveform (dashed red), and high-pass filtered data (solid blue) for a rotate gesture (5). The manual framing timestamps are shown (black), as well as the segmentation timestamps determined by the segmentation algorithm (red). Each of the windows used for feature extraction are also shaded.

Offline Gesture Recognition System

We treated analysis of experimental data as a machine learning classification problem, with three main steps: (1) segmentation, in which we identify the starting and ending timestamps of each gesture, (2) feature extraction, in which we process the raw data to produce separable features for classification, and (3) classification, in which we use a machine learning classifier to decide which gesture in the set of trained gestures is most likely. Here, we describe each of these three parts to our analysis pipeline, which we implemented primarily in Matlab.

Offline Segmentation

In general, segmentation involves identifying a temporally contiguous segment of data in which a gesture of interest has occurred. In this experiment, we controlled the way we collected and marked the data stream. In our data, we know that gestures occurred between framing 'beep' timestamps, since the researcher manually marked those during data collection, and since the participant was in fact reacting to these beeps. However, we do not know exactly when during that period the gesture began and ended, which is important for the way we perform feature extraction and classification. Here, we describe the algorithm used to determine the end points of the gesture.

Figure 3 shows a down-sampled waveform (solid black) of the raw data during a rotate gesture (5) as well as the low-pass filtered DC waveform (dashed red). It is clear from this figure that when the user is at rest (i.e., standing still), the DC is stable, and when the user is performing a gesture (i.e., moving), the DC changes significantly. This observation is the basis of our segmentation algorithm. We deter-

mine the beginning of a gesture by observing when the DC waveform transitions from stable to unstable, and likewise the end of a gesture is when the DC waveform transitions from unstable back to stable.

Before determining the stability of the DC values, the DC waveform itself must be computed. To do so, we first down-sample our raw data which was collected at 250 kS/s by a factor of 1024, resulting a waveform with a sampling rate of 244.14 S/s. To obtain the DC waveform, a 3rd order Butterworth IIR low-pass filter with a 3 dB corner at 10 Hz is applied to the down-sampled data. This filter removes the 60 Hz and higher frequency components and leaves only the DC offset of the data. Figure 3 shows an example of both waveforms. In order to determine the stability of the DC waveform, we divided it into 98 ms windows comprising 24 samples each, and computed two metrics for each window to determine when the user is likely to be moving.

The first metric is based on the interquartile range of the DC waveform. Using this metric, a window is considered to be active (i.e., the user is likely moving during that window) when the interquartile range of the 24 samples in the window is greater than a static threshold of 40 mV. This interquartile range metric identifies windows in which the DC waveform takes on a wide range of values, and therefore is useful in determining when the user is moving.

The second metric is based on the approximate derivative of the DC waveform. In order to remove noise in the DC waveform, we first applied a 3rd order Butterworth IIR low-pass filter with a 3 dB corner at 1 Hz to the DC waveform, and denote the output of this filter f. Next, we compute the finite difference (i.e., sample-to-sample difference) of f, which we call Δf. We consider a window to be active using this metric if the absolute value of the mean of Δf is greater than a dynamically computed threshold. We compute the threshold once for each gesture event to be segmented and used it for all windows within that event. We set the threshold to be 0.6 times the standard deviation of Δf across the entire gesture event, not just across a single window. This metric identifies windows in which the DC value is changing quickly, indicating that the user is moving.

In order to obtain the timestamps of the beginning and end of the gesture, we combine the interquartile range metric and approximate derivative metric by simply taking the logical OR of the two metrics. In other words, if either metric identifies a window as being active, it is considered to be active. This is done because each metric is sensitive to different kinds of variations in the DC waveform, and therefore the OR of the two metrics is needed in order to make the algorithm work across all gestures in the dataset.

The start of the gesture is defined as the first active window within the framing timestamps manually set during data collection. The end of the gesture is the last active window within the manual framing timestamps. Note that it does not matter whether or not all windows between the start and end of the gesture are considered active.

We note that there are a number of limitations to this segmentation approach which we detail later in the description of the real-time segmenter.

Offline Feature Extraction

Once the data for a given gesture is segmented, we extract a number of features to use for classification.

First, we divide our gesture event up into a number of equally spaced windows. We have found that the exact number of windows does not matter much as long as it is greater than about 3. For this experiment we divided the gestures into 5 equally sized windows, and used one window of the same length before the gesture began and one window after the gesture ended, resulting in 7 windows total. The extra window on both ends of the gesture allows the classifier to use the start and end state of the gesture. Figure 3 shows the windows used for feature extraction.

We then compute the same features for each window, both in the time domain, but also in the frequency domain. In the time domain, we compute the DC waveform discussed in the description of the segmentation algorithm. We have observed that the shape of the DC waveform is fairly consistent across many repetitions of the same gesture, and different gestures produce drastically different DC waveforms. For each window we compute the DC value to be the mean of the DC waveform computed for segmentation over the whole window.

It is clear from Figure 3 that the amplitude of the AC signal also changes significantly during a gesture. This is due to the user's body moving closer or farther from electromagnetic noise sources in the environment in addition to changes in the frequency response of the body during a gesture. To obtain a feature which captures these changes, we applied a 3rd order Butterworth IIR high-pass filter with a 3 dB corner at 40 Hz to the same down-sampled data used to generate the DC waveform. This filter removes the DC offset and leaves only the AC signal. This high-pass signal is plotted in Figure 3 (solid blue line). To capture the AC amplitude in this signal, we compute the root-mean-square (RMS) of the high-pass signal over each window.

The DC and RMS features provide a significant amount of information to help classify gestures. However, even more information can be gained by using frequency domain features as well. Since our raw data was collected at a sampling rate of 250 kS/s, we are able to analyze the frequency domain up to the Nyquist frequency of 125 kHz. There are many useful signals throughout this whole frequency band to classify the location of a user, as described in a later section. However, for classifying the gesture being performed, we have found that it is possible to get very high levels of accuracy by using only the frequencies below 500 Hz. This band is dominated by the low-frequency AC power signal (60 Hz in the US) and its harmonics. Since most of the energy radiated off of the power lines and appliances are below 500 Hz, it is not surprising that these features would be the most useful for classification.

In order to compute the frequency domain features below 500 Hz for each window in the gesture, we use the 17 frequencies between 0 and 500 Hz with 30.52 Hz spacing. The 30 Hz resolution allows us to obtain features on and between the peaks in our signal, which is a series of sharp peaks spaced 60 Hz apart (i.e., harmonics of a 60 Hz fundamental). To compute these frequency components, we take an 8192-point fast Fourier transform (FFT) using an 8192-sample Hann window on our raw 250 kS/s data. The 17 frequencies of interest are simply the first 17 FFT bins. In order to take our FFT over arbitrarily sized windows, we compute a number of 8192-point FFTs, each on only 8192 samples of data. To use all samples within a window, we overlap the FFTs such that adjacent FFTs may use the same samples in their computation. The resulting FFT magnitudes are then averaged to produce the equivalent FFT magnitude over the whole window.

We used our two time domain features (i.e., DC and RMS), as well as the 17 frequency domain features in both linear and logarithmic (dB) units. The resulting 36 features per window are concatenated across the 7 windows to produce a feature set of 252 features representing a single gesture.

Offline Classification

In order to determine which gesture was performed, we use the sequential minimal optimization (SMO) implementation of the support vector machine (SVM) found in the Weka machine learning toolkit [6]. An SVM uses labeled data to construct a set of hyper-planes that separate labels in a high-dimensional feature space, which can then be used for classification. Our SVM uses the 252 extracted features to classify which of the 12 gestures had most likely occurred.

Gesture Classification Results

To calculate how accurately we could classify whole-body gestures with our system, we conducted a 10-fold cross-validation on all of our data from each of 2 locations in each of 8 homes and participants.

In each fold, we trained on 36 examples and tested the remaining 4 examples of each gesture. Each fold was made up of data points from a single session of data collection, which in turn consisted of 4 repetitions of each gesture. This ensured that training and testing data points were separated within a fold, and that training data and testing data were separated by several minutes in time, which avoids over-fitting to transient variations in the environment. We assert that these results are representative of what we would expect to see in an interactive system.

The average accuracy across participants and homes was 92.7%, with a standard deviation of 3.0% when classifying between our 12 gestures. This is impressive, given that random chance is about 8.3% for 12 gestures. The maximum aggregate accuracy for a single location was 98.3%, and the minimum was 86.5%.

Close analysis of the confusion matrix (see Figure 4) sheds even more light on the feasibility of Humantenna for classifying whole-body gestures. The most confusion was between the right wave (6) and left wave (7). There was also some confusion between the left arm down (2) and right arm down (3). It is clear that the classifier has the most trouble differentiating left from right, which is because some locations have only small differences in the electromagnetic noise on each side of the body. There was also some confusion between punch twice then kick (11) and kick and then punch twice (12), as these are both very similar gestures. However, there were few other confusions.

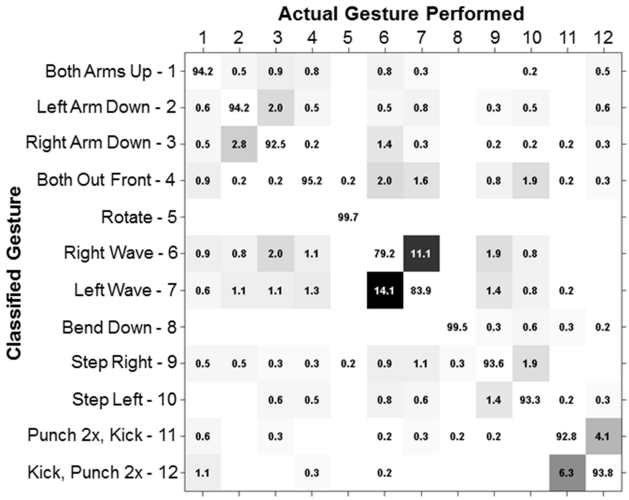

Figure 4: Confusion matrix showing the experiment 1 whole-body gesture classification results. Darker shading indicates more confusion, and the values are percentages.

These results are encouraging and confirm our hypothesis that Humantenna is capable of sensing whole-body gestures by using the body as an antenna receiving electromagnetic noise already present in uninstrumented environments. It further validates our assumption that this would work equally well across a reasonably wide set of people and homes, which is also very important.

EXPERIMENT 2: CLASSIFYING LOCATION

Previous work using the human body as an antenna has shown the ability to robustly classify the user's location in a home while the user is standing still near a wall [3]. To verify the feasibility of classifying the user's location in more realistic settings, when the user is away from the wall and performing gestures, we conducted a second, smaller study.

Experimental Procedure

Three participants took part in this study, which was run in two different homes. We ran a pair of participants in each home in a single day, with one participant in common between the two homes. Within each home, we chose 8 different locations. Two of the locations were in the same large room, and the other rooms were distributed throughout the home. We chose locations that varied in terms of the amount of open space as well as the number and type of electronic appliances present. Each participant performed gestures in 5 of the 8 chosen locations. This meant that 2

locations were shared between each pair of participants, allowing us to examine stability across people.

The participants performed only a subset of 8 gestures from the full set of 12 shown in Figure 2: (1) both arms up, (2) left arm down, (3) right arm down, (5) rotate, (6) right hand wave, (7) left hand wave, (9) step to the right, and (10) step to the left. This gesture set contains mostly arm waves on both sides of the body since these are the types of gestures which our gesture classifier produces the most confusions.

We used the same equipment and procedure as in the first experiment for whole-body gesture sensing. However, we only performed 5 sessions, resulting in 20 examples of each gesture rather than 40. Therefore, each participant completed a total of 800 whole-body gestures in each home.

Offline Location Classification System

Unlike gesture classification, it is not necessary to capture how the signal changes over time in order to determine the location of the user. Therefore, instead of running a segmentation algorithm to find the start and end of each gesture, and then dividing the gestures into discrete windows, we instead use a single window at the beginning of the gesture. We simply take the first 0.5 s after the beginning framing timestamp that was manually set by the observer as our window from which to extract features.

We used the same DC and RMS features as in the gesture classification, computed in the exact same manner. For the frequency domain, we again computed an 8192-point FFT using an 8192-sample Hann window on our raw 250 kS/s data. However, past work using the human body as antenna for location classification suggests much of the differentiating signal for determining the location of a user comes from the presence of high frequency peaks which are radiated from appliances in the home [3]. Therefore, instead of using only frequency bins below 500 Hz, we use the magnitude of all frequency bins below 4 kHz, as this is where the majority of the harmonics of 60 Hz are found. We also run a moving average across the frequency bins, with the window size of 1 kHz, and extract features at 1 kHz intervals across the whole frequency band, from 0 to 125 kHz.

Instead of using the absolute magnitude from each frequency bin, we normalize all frequency domain features by the amplitude of the 60 Hz peak, which is the fundamental frequency of the signal emitted from the power lines. In the 0 to 4 kHz band, we use 132 relative frequency domain features at a 30.52 Hz spacing. In the full 0 to 125 kHz band, 129 features are used with approximately 1 kHz spacing. Since we again use all frequency domain features in linear and logarithmic (dB) units, we have a total of 522 frequency domain features, plus 2 time domain features, resulting in 524 total features.

Location Classification Results

To test our ability to classify the user's location, we ran 5-fold cross-validations, with each fold made up of data points from a single session, as was done for gesture classification. Again, folding by session provides an accurate representation of what is expected in an interactive system.

Running the cross-validation for each participant in each home, we obtain an average classification accuracy of 99.6% (σ = 0.4%) when classifying between the 5 locations used by each participant (chance = 20%).

These high classification accuracies reinforce results seen in the previous work using the human body as an antenna for user location classification [3]. However, since we are able to classify whole-body gestures using a sampling rate of only 976.56 S/s, it would be much more practical for many applications if the location could also be classified with this low sampling rate.

To test this, we ran all of our cross-validations again using the same feature set used in our gesture classification experiment (i.e., only 36 features). This results in an average classification accuracy of 97.1% (σ = 2.1%). This accuracy is lower than what we obtained using the much larger feature set, but it still a very high accuracy and is probably good enough for many applications. The advantage of this approach is that user location and gesture classification can be done using the same features, which we demonstrate can be extracted in real-time in the next section.

One potential caveat when using this reduced feature set is that the classifier is not able to take advantage of the high frequency peaks produced by many of the appliances at different locations. As a result, it must use only the differences in the strength of 60 Hz wave and its harmonics to fingerprint each location. As a result, such a feature set will probably not be as robust to changes in the electrical state of the home as well as temporal drift, but verifying this and exploring solutions to it remains future work.

Since the location classifier uses differences in the strengths of certain frequencies to fingerprint a location, we hypothesized that it should work well across users. To test this, we trained our classifier on one user, and tested it on the data from the other user in each of the 2 shared locations in both homes. We achieve an accuracy of 100% using the full feature set, and 96.3% using the reduced feature set described above. However, since there were only 2 locations shared between users, random chance is much higher at 50%.

We thus applied a more stringent test by training the 5-location classifier using one participant and then testing the 2 locations they had in common using the other participant. This results in 96.1% accuracy using the full feature set, and 84.6% accuracy using the reduced feature set. This suggests that models built off users could be generalized relatively well to other users.

Similarly, we decided to test whether we could classify the home and location of the user from the data collected in our main gesture experiment. In this case, we have 8 homes and 2 locations in each home, and thus chance is 6.25% (i.e., 16 classes). We obtained an accuracy of 99.4% using the full feature set, and 94.1% using the reduced feature set.

Thus, in addition to being able to classify the whole-body gestures being performed, we are also able to classify the user's location with a very high level of accuracy. In addition, this classification can be done using the same feature set used for gesture classification, while experiencing only a small decrease in classification accuracy. We have also shown that the location can be classified independently of the user, and therefore a single user can train a location classifier which works well for all other users. Although these results are promising, they come from a very small study involving only a few different locations, and therefore a larger study is necessary to verify the results in general.

REAL-TIME INTERACTIVE SYSTEM

The experiments demonstrate the feasibility of using the human body as an antenna for sensing and classifying whole-body gestures as well as location. Building on those results, we describe the extensions that we developed in order to turn the offline processing methodology into a real-time whole-body recognition system.

Real-Time Data Capture

In the offline experiments, we collected data at the maximum sampling rate of 250 kS/s, sent it over Wi-Fi to a computer and logged it to a hard disk for later post-processing. In a real-time system, the data needs to be processed immediately after it is captured from the data acquisition unit. Fortunately, our experiments taught us that we only needed the frequency components of the signal up to 500 Hz to perform highly accurate whole-body gesture classification. As a result, our real-time system only samples data at a rate of 976.56 S/s, which is 256 times less than the rate used for the offline experiments. The lower sampling rate greatly reduces the required hardware and software specifications for the real-time system. The captured data is buffered by the data acquisition hardware into 32 sample buffers, meaning that a new buffer is captured every 33 ms. All of the remaining processing described in the following sub-sections is computed per 33 ms frame.

Real-Time Segmentation

The segmentation algorithm used in the gesture sensing experiment cannot run in real-time because it relies on the presence of the manual framing timestamps recoded during data collection. A real-time segmenter must automatically identify gesture events from the live data stream. In addition, minimizing latency is important because the segmented event must be identified before the classifier can run.

To segment gestures in real-time, we first down-sample the captured data by 4 from 976.56 S/s to 244.14 S/s, which is the same sampling rate used by the offline segmenter. We apply a 3rd order Butterworth IIR low-pass filter with a 3 dB corner at 1.5 Hz to this data to obtain f. The 3 dB corner of this filter was moved from 1 Hz to 1.5 Hz in the real-time implementation in order to reduce the latency of the segmentation. With the corner at 1 Hz, the group delay of the filter is 320 ms (78 samples); however, moving the corner to 1.5 Hz reduces the group delay to 213 ms (52 samples). Next, we compute Δf by taking the finite difference (i.e., sample-to-sample difference) of f. If the absolute value of the mean of Δf is greater than a static threshold, then we consider the frame to be active.

We then apply smoothing to remove noise. First, we consider any periods of inactivity less than 197 ms (6 frames) to be active. This essentially removes small sections of inactivity in the computed activity metric. Next, we remove any period of activity less than 1.02 s (31 frames) because gestures are assumed to be at least 1 second long. Any resulting event in the smoothed activity metric is considered to be a gesture to be classified.

As a result of the 213 ms group delay of the low-pass filter and the 197 ms wait period to check for another period of activity, the latency of the real-time segmentation algorithm is 410 ms. Reducing this latency remains future work.

Real-Time Feature Extraction

We perform the real-time feature extraction in two parts in order to reduce latency. First, we compute the DC, RMS, and FFT values per frame in parallel with the segmentation (i.e., in a separate thread). For the time domain features, we down-sample the streaming data by 4 so that the sampling rate is 244.14 S/s, and then compute the DC and RMS features in the same way as in the offline segmenter. We obtain the 17 frequency domain components between 0 and 500 Hz with a spacing of 30.52 Hz by taking a 32-point FFT using a 32-sample Hann window on each 32 sample frame. The DC, RMS, and FFT features are computed for each frame and are queued until a gesture is segmented.

Once the segmenter finds a gesture, we dequeue the extracted features corresponding to all data frames during the gesture. We divide the data frames into 5 equal sized windows, and compute the average DC, RMS, and FFT magnitude values over each window. We compute the 17 FFT magnitude features in dB as well as linear units. Like the offline features, the real-time system uses 36 features per frame (i.e., DC, RMS, 2x17 FFT bins). Since the real-time system uses only 5 windows per gesture, with no windows before or after the gesture, a total of 180 features are used.

Real-Time Classifier

To classify the segmented gesture, all 180 features are then fed to an SMO implementation of the SVM classifier found in the Weka machine learning toolkit [6]. The SVM runs the new features against a previously trained model to produce a classification result, which is sent the user interface.

Demonstration Applications

We have implemented two simple applications to demonstrate the Humantenna real-time interactive system; both are demonstrated in the attached video figure.

In one application, the user performs a gesture, which is recognized and mirrored by an appropriate pre-recorded stick figure on screen. The other application we developed is a game of Tetris in which the user controls the falling blocks with whole-body gestures. Although any gesture can be trained, we used step right and step left to move the

block right and left respectively. To rotate the block we performed a rotation gesture with both hands, as if the user was picking up a physical block and rotating it. To drop the block to the bottom of the screen the user would stomp their left foot. Figure 5 shows an image of a user playing Tetris using the Humantenna demonstration.

In both demonstration applications, we trained the classifier using 20 examples of each gesture, and obtained classification accuracies of about what we would expect using the offline system (i.e., low 90%). We also learned from our interactive system that changes in the environment make the use of a static threshold for segmentation somewhat brittle over longer periods of time. In order to build a segmenter that is more robust to environmental changes, a computed dynamic threshold should be used, as it was in the offline system. Doing this remains future work.

DISCUSSION AND FUTURE WORK

Through the experiments presented in this paper, we have shown that using the human body as an antenna, the Humantenna system can classify whole-body gestures at about 93% accuracy, and can classify the user's location at almost 100% accuracy. In addition, we have demonstrated the ability to run the Humantenna system in real-time. This section discusses the size of the training set needed to achieve this accuracy, the limitations of the current system, and future work to improve the system.

Size of Training Set

The results presented earlier in this paper were obtained using a classifier that was trained using either 16 or 36 training examples. For many practical applications of Humantenna is it important to reduce the size of this training set as much as possible. We therefore conducted a simple experiment in which we ran all of our cross-validations again while varying the number of sessions of data used for training the classifier. We found that the average accuracy across all participants in all locations seemed to converge when the training set contained at least 16 examples of each gesture. However, even with the size of the training set as low as 4 examples of each gesture, we achieved gesture classification accuracies of 84.5%.

This indicates that a simple impromptu application of Humantenna can be quickly trained and still achieve relatively high levels of accuracy. When more accuracy is needed, additional training examples can be added. In addition, with enough training examples, it may be possible to train a generalized model which works well for a variety of users, homes, and locations.

Ultimately, we envision using an incremental machine learning approach, in which the generalized model is used as a baseline, and a personalized model is created as a person uses the system in a new location. Such an approach would also allow the model to adapt to changes in the environment as well as changes in how a user performs a gesture over time.

Figure 5: User playing a Tetris game using the Humantenna real-time whole-body gesture sensing system.

Limitations and Future Work

Our current classification approach divides each gesture into a fixed number of windows, computes features, and uses the aggregate feature vector in an SVM classifier. While this windowing approach scales reasonably well to performing gestures at different speeds, it requires sub-parts of the gestures to have the same relative timing. For example, in the gesture where the user punches twice and kicks, the classifier will only be able to handle changes in the speed of this gesture if all parts of the gesture's speed are scaled equally.

In addition, this approach requires that the segmentation algorithm be stable and provide very precise timestamps of the beginning and end of each gesture. If these start and end positions are shifted even slightly, the windows will be shifted and thus the classifier is likely to misclassify. This worked surprisingly well in our experiments, but in future work, we propose instead using a classifier that considers discrete states of each gesture rather than windows in time, for example a hidden Markov model (HMM), or conditional random field (CRF).

Furthermore, an HMM, or similar approach may also be beneficial in reducing the latency of the system. The current segmentation algorithm has high latency because it cannot identify the occurrence of a gesture until a fixed period after the event has ended. With an HMM, the segmentation could be implemented as a threshold on the likelihood that the current sequence of events represents a valid gesture. This would allow gesture segmentation and identification to occur while the gesture is being performed rather than some time after it has completed.

Another important question is which locations in a building work better than others. Through our experimentation, we have found significant variation in ability to classify gestures at different locations. Thus far, we have not been able to test the system at a large enough number of locations to obtain statistically significant conclusions about what prop-

erties of a location makes it good or bad for the Humantenna system. This detailed exploration remains future work.

As with previous work, the electrical state of the home (i.e., which appliances and lights were turned on) remained relatively constant throughout our experimental sessions. In any real deployment, it is reasonable to expect lights and other appliances to be turned on and off during the use of the system. Fully exploring the robustness of our system to these changes remains future work, although we suggest several approaches for dealing with the issues. First, many of these variability issues can be handled with a larger training set which includes examples from multiple electrical states of the home. In addition, we intend to explore a feature set which is more robust to changes, as we have already done for the location classification (i.e., using normalized frequency domain features rather than absolute magnitudes). Additionally, it is possible to monitor the state of the home using existing systems [4], and therefore change the classifier's model based on the sensed state.

We also plan to explore the ability to implement Humantenna on a small mobile device located in a user's pocket, for example a cellular phone. The sensing hardware is simply an analog-to-digital converter running at around 1 kS/s and a large resistor to provide a DC bias. Therefore, it is easy to implement the required hardware on any current mobile device. However, it remains future work to explore the possibility of sensing the electromagnetic signal received by the human body antenna without physical contact to the body. In this case, we would need to use a short range air-coupled connection to the body instead.

Although the detailed experiments conducted in this paper focus on gesturing in the home, we believe that Humantenna will work equally well in commercial buildings and other environments. We informally conducted experiments in a modern office building, and although the signal received by the body is significantly different from the signal seen in a residential environment, the segmentation and gesture classification appeared to work just as well. We plan to further explore the ability to use the human body as antenna for sensing gestures in other environments other than the home.

CONCLUSION

By extending past work using the human body as an antenna for recognizing touch gestures, we have built a real-time system to sense whole-body gestures. This system could allow truly mobile and ubiquitous whole-body interaction by eliminating the need for instrumenting the interaction environment. We have shown the ability of our system to sense a user's whole-body gestures with an average accuracy of 93%, as well as classify the user's location within a building at nearly 100% accuracy. We also implemented our whole-body sensing system in real-time to demonstrate its ability to operate as an interactive user input system. This work suggests the feasibility of building real-time, interactive whole-body gesture sensing systems on mobile platforms carried by a user, and thus enabling a variety of applications of whole-body interaction.

ACKNOWLEDGEMENTS

We thank Greg Smith for implementing the user interface for the interactive demonstration application, and Asta Roseway for drawing Figure 1.

REFERENCES

1. Agrawal, S., Constandache, I., Gaonkar, S., Choudhury, R., Caves, K, DeRuyter, F. Using mobile phones to write in air. *In Proc of Mobisys 2011*, 15-28.
2. Chen, W.T. and Chuang, H.R. Numerical Computation of the EM Coupling between a Circular Loop Antenna and a Full-Scale Human-Body Model. *IEEE Trans. Microw. Theory Tech.,* 46.10 (1998), 1516-1520.
3. Cohn, G., Morris, D., Patel, S.N., Tan, D.S. Your Noise is My Command: Sensing Gestures Using the Body as an Antenna. In *Proc of ACM CHI 2011*, 791-800.
4. Gupta, S., Reynolds, M.S., Patel, S.N. ElectriSense: Single-Point Sensing Using EMI for Electrical Event Detection and Classification in the Home. In *Proc of ACM Ubicomp 2010*, 139-148.
5. Hall, P.S. and Hoa, Y. Antennas and Propagation for Body Centric Communications. *Proc Euro Conf on Antennas and Propagation 2006*, 1-7.
6. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H. The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11.1 (2009).
7. Harrison, C., Tan, D. Morris, D. Skinput: Appropriating the Body as an Input Surface. In *Proc of CHI 2010.*
8. Junker, H., Lukowitz, P., Troester, G. Continuous recognition of arm activities with body-worn inertial sensors. In *Proc of ISWC 2004*, 188-189.
9. Larson, E., Cohn, G., Gupta, S., Ren, X., Harrison, B., Fox, D., Patel, S.N. HeatWave: thermal imaging for surface user interaction. In *Proc of CHI 2011*, 2565-2574.
10. Michoud, B., Guillou, E., Bouakaz, S. Real-time and markerless 3D human motion capture using multiple views. In *Proc of Human Motion 2007*, 88-103.
11. Rekimoto, J. GestureWrist and GesturePad: Unobtrusive Wearable Interaction Devices. In *Proc of ISWC 2001.*
12. Saponas, S., Tan, D., Morris, D., Balakrishnan, R., Turner, J., Landay, J. Enabling Always-Available Input with Muscle-Computer Interfaces. In *Proc of ACM UIST 2009*, 167-176.
13. Scott, J., Dearman, D., Yatani, K., Truong, K.N. Sensing foot gestures from the pocket. In *Proc of UIST 2010*.
14. Vicon. http://www.vicon.com
15. Wachs, J., Kölsch, M., Stern, H., Edan Y. Vision-based hand-gesture applications. In *Comm. ACM* 54.2, 60-71.
16. Xbox Kinect. http://www.xbox.com/en-US/kinect.