It's the Human that Matters: Accurate User Orientation Estimation for Mobile Computing Applications

Nesma Mohssen  
Computer and Systems Eng. Dept.  
Faculty of Engineering, Alexandria Univ., Egypt  
nesma.mohssen.k@alexu.edu.eg

Heba Aly  
Computer and Systems Eng. Dept.  
Faculty of Engineering, Alexandria Univ., Egypt  
heba.aly@alexu.edu.eg

Rana Momtaz  
Computer and Systems Eng. Dept.  
Faculty of Engineering, Alexandria Univ., Egypt  
rana.momtaz@alexu.edu.eg

Moustafa Youssef  
Wireless Research Center  
E-JUST, Egypt  
moustafa.youssef@ejust.edu.eg

ABSTRACT

Ubiquity of Internet-connected and sensor-equipped portable devices sparked a new set of mobile computing applications that leverage the proliferating sensing capabilities of smartphones. For many of these applications, accurate estimation of the user heading, as compared to the phone heading, is of paramount importance. This is of special importance for many crowd-sensing applications, where the phone can be carried in arbitrary positions and orientations relative to the user body. Current state-of-the-art focus mainly on estimating the phone orientation, require the phone to be placed in a particular position, require user intervention, and/or do not work accurately indoors; which limits their ubiquitous usability in different applications.

In this paper we present Humaine, a novel system to reliably and accurately estimate the user orientation relative to the Earth coordinate system. Humaine requires no prior-configuration nor user intervention and works accurately indoors and outdoors for arbitrary cell phone positions and orientations relative to the user body. The system applies statistical analysis techniques to the inertial sensors widely available on today’s cell phones to estimate both the phone and user orientation. Implementation of the system on different Android devices with 170 experiments performed at different indoor and outdoor testbeds shows that Humaine significantly outperforms the state-of-the-art in diverse scenarios, achieving a median accuracy of 15° averaged over a wide variety of phone positions. This is 558% better than the state-of-the-art. The accuracy is bounded by the error in the inertial sensors readings and can be enhanced with more accurate sensors and sensor fusion.

1. INTRODUCTION

Recent advances in ubiquitous computing highlighted the importance of user direction estimation; it enables plentiful ubiquitous and mobile computing applications such as localization [3, 4, 7, 8, 24], activity recognition [28], virtual reality, among others [1, 6, 21]. Today’s smart-phones are equipped with a number of inertial sensors, e.g. accelerometer, magnetometer, and gyroscope. These sensors provide a measurement for the cell phone orientation relative to the magnetic North. However, users carry cell phones in arbitrary positions and orientations that are typically disoriented with respect to the user (Figure 1). Depending on the phone orientation, rather than the user orientation, can lead to huge errors in many applications. For example, in the popular dead-reckoning localization techniques, e.g. [7, 26], the user displacement obtained from the inertial sensors is combined with the movement direction to estimate the next user location. Erroneous direction estimation, just based on the returned phone orientation from the OS API, will result in a large error in localization that accumulates quickly with time. This is especially true for the growing field of mobile applications that do not assume a certain placement or orientation of the phone, e.g. in hand or in pocket, such as crowd-sensing applications.

To overcome these problems and obtain the actual user direction, a number of systems have been proposed [5, 13, 14, 18, 19, 22, 25]. For example, [5, 18, 22, 25] depend on special external sensors attached to the user’s leg or placed in her pants pocket to detect her orientation. Other systems, e.g. [13, 14, 19], use the inertial sensors in standard cell phones to estimate the user direction. However, these solutions either work only in specific phone positions (e.g. pants pocket), are limited to outdoor environments where the phone sensors are not affected by the indoor magnetic noise [13, 14], or require special user actions (e.g. [19]). These limit their ubiquitous usability.

In this paper, we present Humaine: a system capable of accurately estimating the user orientation in different environments at arbitrary phone positions and orientations without user intervention. Humaine starts by fusing the different phone inertial sensors to obtain the phone orientation in the user horizontal plane of motion. To obtain the actual user heading, Humaine takes advantage of the observation that the direction of motion is the direction that has the maximum acceleration variance. Therefore, it applies the principal component analysis (PCA) on the linear ac-
phone API that uses the magnetometer or they depend on other phone sensors, e.g. the camera [23], to obtain more accurate results. However, even with zero error in phone direction estimation, using phone-heading estimates can lead to huge errors in case the phone is not aligned with the user direction. Throughout this section, we discuss the different techniques for user direction estimation that have been proposed in literature.

2.1 Techniques based on Location Estimation

Frequent logging of user location, e.g. using GPS [12], can provide accurate user direction. In this case, the direction is estimated as the direction of the line joining the last two estimated locations. However, this depends on the availability and the accuracy of the localization system. For example, since the GPS accuracy is low in urban areas [9], the direction estimation is not accurate. In addition, it completely fails in indoor environments and has significant latency and energy-consumption as we quantify in Section 4.

2.2 Techniques that use Special Sensors

To overcome this limitation, researchers employed other sensors (e.g. wearable cameras, shoe-mounted sensors, and/or inertial head trackers) to provide accurate direction estimation [5, 15–18, 22, 25]. In [16, 18, 22], a special acceleration sensor, e.g. the MTx motion sensor [18] is used to obtain the rotation axis that is parallel to the movement direction, but it fails to determine the forward direction itself, leading to a “180° ambiguity problem”. [18] addressed the 180° ambiguity problem using integration of the acceleration signal in the global frame, which does not yield a robust estimate [22]. To detect the forward direction, [22] leverages the rotational motion of the sensor before foot impact. However, using the foot impact to solve the 180° ambiguity leads to a problem when applying the technique indoors as we show in Section 4.

In [16], authors used inertial sensors, a wearable camera, and an inertial head tracker. The forward direction is determined by testing whether the slope of vertical acceleration at the peak for forward acceleration is increasing. This algorithm requires sensors to be attached to the torso for correct detection of the acceleration patterns, limiting its applicability for different positions. Also, using a wearable camera imposes further limitations on the applicability of the technique and the environment (e.g. lightning).

In [16], authors provide a thorough study on the effect of different phone positions (in hand, in pants pocket, in bag, in shirt pocket), phone orientations relative to the user body, and indoor/outdoor effect on the different user direction estimation techniques.

The rest of the paper is organized as follows: In Section 2, we discuss related work. Section 3 gives the details of the Humaine system. We provide the implementation and evaluation of the system in Section 4. Finally, Section 5 concludes the paper and gives directions for future work.

2. RELATED WORK

Applications that require the user heading direction estimation, e.g. [7, 26], usually assume that the phone is oriented with the user. They either depend on the standard cell phone coordinate system $(X, Y, Z)$ or the world coordinate system $(F, S, −G)$, which in turn is misaligned with the user coordinate system $(N, E, −G)$. The user plane of motion is the $(F, S)$ plane, which is the plane perpendicular to gravity.

Figure 1: The Different coordinate systems: The phone coordinate system $(X, Y, Z)$ is misaligned from the human coordinate system $(F, S, −G)$, which in turn is misaligned with the world coordinate system $(N, E, −G)$. The user plane of motion is the $(F, S)$ plane, which is the plane perpendicular to gravity.

In summary, our main contributions are three-fold:

- We present the architecture and details of Humaine: a system to detect accurate user orientation using standard cell phones at arbitrary positions and orientations in both indoor and outdoor environments.
- We implement our system on Android-based mobile devices and evaluate its performance as compared to state-of-the-art systems.
- We provide a thorough study on the effect of different phone positions (in hand, in pants pocket, in bag, in shirt pocket), phone orientations relative to the user body, and indoor/outdoor effect on the different user direction estimation techniques.

The rest of the paper is organized as follows: In Section 2, we discuss related work. Section 3 gives the details of the Humaine system. We provide the implementation and evaluation of the system in Section 4. Finally, Section 5 concludes the paper and gives directions for future work.
The uDirect system [13,14] employs a similar technique to estimate the user direction; They identify a point in the middle between the user’s detected heel strike and toe-off moments as the point where the device orientation is close to the device orientation in the standing mode. These systems, however, require a model for the acceleration pattern within a step for each phone position; A model for the phone placed in the pants pocket was presented in [13,14]. Deriving the model for a new position is not straightforward and the acceleration pattern for other positions may not be as clear as in the case of the presented pants pocket. Furthermore, the magnetic field noise, which affects the oriented/rotated acceleration pattern used in heading estimation, degrades uDirect accuracy indoors as we quantify in Section 4.

2.4 Summary

Table 1 summarizes the differences between Humaine and the most relevant state-of-the-art. The current state-of-the-art either require special external sensors, work in a specific phone position (e.g. in pants pocket) or orientation (e.g. oriented phone with the user), require user intervention, and/or work only outdoors.

Humaine, on the other hand, depends on available inertial sensors in standard cell phones. It is applicable indoors and outdoors for arbitrary phone positions and orientations without any intervention from the user.

3. THE HUMAINE SYSTEM

In this section, we provide the details of the different components of the system; covering the sensor readings preprocessing, the acceleration transformation to the user’s plane, detection of the user’s motion axis, and finally detection of the user’s orientation relative to North. We start by defining the coordinate systems used in the paper, followed by an overview of the system architecture, and finally the details of each module.

3.1 Coordinate Systems

Figure 1 shows the different coordinate systems used in the paper. The world coordinate system \((N,E,-G)\) is defined by North \((N)\), East \((E)\), and the Earth gravity \((-G)\). We refer to the phone local coordinate system as \((X,Y,Z)\). The user plane of motion is perpendicular to gravity \((-G)\) and we are interested in the user forward direction \((F)\). Therefore, the user coordinate system is defined as \((F,S,-G)\) where \(S\) points toward the right side of the user’s forward direction \((F)\). Table 2 summarizes all symbols used in this section.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(N,E,-G)</td>
<td>The world axes (Figure 1): (N) points to North, (E) points to East, and (-G) is opposite to gravity.</td>
</tr>
<tr>
<td>(F,S,-G)</td>
<td>The axes of the user coordinate system (Figure 1): (F) points in the forward user motion direction, (S) points toward the side, and (-G) is opposite to gravity.</td>
</tr>
<tr>
<td>(X,Y,Z)</td>
<td>The axes of the phone coordinate system (Figure 6): (X) points toward the phone side, (Y) points toward the phone head, and (Z) is perpendicular to the phone screens.</td>
</tr>
<tr>
<td>(\alpha)</td>
<td>The phone pitch angle (Figure 6).</td>
</tr>
<tr>
<td>(\beta)</td>
<td>The phone roll angle (Figure 6).</td>
</tr>
<tr>
<td>(\gamma)</td>
<td>The phone yaw angle relative to North (phone azimuth, Figure 6).</td>
</tr>
<tr>
<td>(\theta_u)</td>
<td>The user orientation angle relative to North.</td>
</tr>
<tr>
<td>(R_A(b))</td>
<td>Rotation matrix of angle (b) around axis (A).</td>
</tr>
<tr>
<td>(g)</td>
<td>The Earth’s gravity acceleration ((9.80665m/s^2)).</td>
</tr>
<tr>
<td>(a_t)</td>
<td>Total acceleration affecting the phone in the phone coord. system (Figure 3).</td>
</tr>
<tr>
<td>(\ell)</td>
<td>Linear acceleration in the phone coordinate system (Figure 3).</td>
</tr>
<tr>
<td>(\ell_u)</td>
<td>((\ell_F,\ell_S)), the linear acceleration in the user coordinate system.</td>
</tr>
<tr>
<td>(\psi)</td>
<td>The dip angle of the magnetic field measured downwards from the horizontal plane [20] for a perfectly oriented phone, i.e. pointing to North.</td>
</tr>
<tr>
<td>(G_m)</td>
<td>([g_x, g_y, g_z]), gravity acceleration components in the phone coordinate system (Figure 3).</td>
</tr>
<tr>
<td>(M)</td>
<td>([m_x, m_y, m_z]), the measured magnetic field in the phone coordinate system.</td>
</tr>
<tr>
<td>(\delta)</td>
<td>Noise filter smoothing factor.</td>
</tr>
<tr>
<td>(\omega)</td>
<td>PCA window size.</td>
</tr>
</tbody>
</table>

Table 2: Summary of the symbols used.
readings to reduce the noise.

The User Direction Estimation Module transforms the linear acceleration readings to the user plane of motion and estimates the user motion axis as the direction with the maximum variance. Finally, the Ambiguity Resolution Module disambiguates the final user’s heading direction along the motion axis.

3.3 Sensor Fusion

The system collects raw sensor information from inertial sensors available in the cell phone. In particular, we collect the 3D acceleration, the 3D magnetic field from the magnetometer, and the relative rotation angle from the gyroscope. Sensor fusion is an important step to reduce noise and estimate the quantities of interest. Figure 4 shows an overview of our sensor fusion module.

3.3.1 Gravity Acceleration Estimation

The accelerometer returns the acceleration force applied to the phone at any time instance is the sum of the gravity acc. \( g \) and the linear acc. \( \ell \). The gravity acc. components in the phone coordinate system are \((g_x, g_y, g_z)\).

Users carry cell phones in different positions (e.g. pants’ pocket, shirt’s pocket, in bag, or in hand). Therefore, the phone can have an arbitrary orientation relative to the user’s plane of motion. To estimate the phone orientation, we depend on the estimated gravity acceleration (Section 3.3.1). For ease of illustration, we first use Euler angles and rotation
where \( R \) with angle rotation matrices.

Figure 6: Different phone orientation angles: pitch \((\alpha)\), roll \((\beta)\), and yaw \((\gamma)\). \( \gamma \) is the phone orientation angle relative to North (azimuth angle).

Figure 7: The phone orientation after applying the rotation matrices \( R(-\beta)R(-\alpha) \). The phone \( Z \)-axis becomes aligned with \(-G\) and the phone rests in the horizontal plane of motion making an angle \( \gamma \) with North (phone orientation angle).

matrices to explain how Humaine estimates the phone orientation angles around the three main axes; i.e. yaw, pitch, and roll (Figure 6). Then, we explain our actual implementation.

Let \( G_m = [g_x, g_y, g_z]^T \) be the gravity components, of the estimated gravity acceleration affecting the phone in its arbitrary location. \( G_m \) can be written as:

\[
\begin{bmatrix}
g_x \\
g_y \\
g_z \\
\end{bmatrix} = R_A(b) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}
\]

where \( R_A(b) \) is the rotation matrix representing a rotation with angle \( b \) around axis \( A \). In particular:

\[
R_X(\alpha) = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos(\alpha) & \sin(\alpha) \\
0 & -\sin(\alpha) & \cos(\alpha)
\end{bmatrix}
\]

\[
R_Y(\beta) = \begin{bmatrix}
\cos(\beta) & 0 & -\sin(\beta) \\
0 & 1 & 0 \\
\sin(\beta) & 0 & \cos(\beta)
\end{bmatrix}
\]

\[
R_Z(\gamma) = \begin{bmatrix}
\cos(\gamma) & \sin(\gamma) & 0 \\
-\sin(\gamma) & \cos(\gamma) & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

To obtain the pitch \((\alpha)\) and roll \((\beta)\) angles, we multiply both sides of Equation 1 by \( R_A^{-1}(\beta)R_A^{-1}(\alpha) \) and noting that \( R_A^{-1}(b) = R_A(-b) \), we obtain:

\[
R_Y(-\beta)R_X(-\alpha)G_m = R_Z(\gamma) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}
\]

Substituting from equations 2, 3, and 4 we obtain:

\[
\begin{bmatrix}
\cos(\beta) & \sin(\beta) & 0 \\
0 & \cos(\alpha) & \sin(\alpha) \\
-\sin(\beta) & \cos(\beta) & 0
\end{bmatrix} \begin{bmatrix}
g_x \\
g_y \\
g_z
\end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}
\]

From the second and first rows of Equation 6 we get:

\[
\tan(\alpha) = \frac{g_y}{g_x}
\]

\[
\tan(\beta) = \frac{g_z}{g_y \sin(\alpha) + g_x \cos(\alpha)}
\]

To obtain the yaw angle \((\gamma)\), which is the phone orientation angle relative to North (Figure 7), we leverage the magnetometer signal. In particular, let \( M = [m_x, m_y, m_z]^T \) be the magnetic field at the arbitrary phone orientation. Using a similar approach to Equation 5, we get

\[
\begin{bmatrix}
m_x \cos(\beta) + m_y \sin(\beta) \sin(\alpha) + m_z \sin(\beta) \cos(\alpha) \\
m_y \cos(\alpha) - m_z \sin(\alpha) \\
-m_x \sin(\beta) + m_y \cos(\beta) \sin(\alpha) + m_z \cos(\beta) \cos(\alpha)
\end{bmatrix} = \begin{bmatrix} E \cos(\psi) \sin(\gamma) \\ -E \cos(\psi) \cos(\gamma) \\ E \sin(\psi) \end{bmatrix}
\]

where \( E \) and \( \psi \) are the unknown Earth magnetic field strength and the dip angle of the field measured downwards from horizontal [20] for a perfectly oriented phone, i.e. pointing to North. From the first and second rows of Equation 9 we get:

\[
\tan(\gamma) = \frac{m_x \cos(\beta) + m_y \sin(\beta) \sin(\alpha) + m_z \sin(\beta) \cos(\alpha)}{m_x \sin(\alpha) - m_y \cos(\alpha)}
\]

Equations 7, 8, and 10 are solved taken into account the sign of the gravity acceleration and magnetometer components to obtain the correct quadrant for the respective angles.

**Actual Implementation**

We use unit quaternions to represent the rotation operation in Humaine rather than rotation matrices. This representation evolves from Euler’s rotation theorem [10], which implies that any rotation or sequence of rotations of a rigid body in a three-dimensional space is equivalent to a pure rotation about a single fixed axis \( r = [r_x, r_y, r_z]^T \), \( r_x^2 + r_y^2 + r_z^2 = 1 \) by an angle \( \theta \) (Figure 8). Quaternions give a simple way to encode the rotation operation in 4 parameters, compared to 9 parameters in case of rotation matrices. In addition, quaternions suffer from fewer mathematical rounding defects and are not subject to the gimbal lock problem [10, 11].

The relation between the pitch, roll, and yaw angles obtained in the previous section and the 4D quaternion vector
Where s linear acceleration (e.g., due to shaking). We apply a low-pass filter to these measurements, which are sensitive to abrupt changes in the linear acceleration vector from raw sensor measurements; this is as follows:

\[
q = \begin{bmatrix}
\cos\left(\frac{\theta}{2}\right) \\
\sin\left(\frac{\theta}{2}\right) r_x \\
\sin\left(\frac{\theta}{2}\right) r_y \\
\sin\left(\frac{\theta}{2}\right) r_z
\end{bmatrix}
\]

\[
= \begin{bmatrix}
\cos\left(\frac{\gamma}{2}\right) \cos\left(\theta\right) \cos\left(\theta\right) + \sin\left(\frac{\gamma}{2}\right) \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right) \\
\sin\left(\frac{\gamma}{2}\right) \sin\left(\theta\right) - \sin\left(\frac{\gamma}{2}\right) \cos\left(\theta\right) \cos\left(\theta\right) \\
\sin\left(\frac{\gamma}{2}\right) \cos\left(\theta\right) + \cos\left(\frac{\gamma}{2}\right) \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right) \\
\cos\left(\frac{\gamma}{2}\right) \sin\left(\theta\right) - \sin\left(\frac{\gamma}{2}\right) \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right)
\end{bmatrix}
\]

This is equivalent to a single rotation around axis \(\vec{r}\) with an angle \(\theta\).

3.4 Preprocessing

Humaine estimates the user direction based on processing the linear acceleration vector from raw sensor measurements; these measurements are sensitive to abrupt changes in the cell phone, e.g., due to shaking. We apply a low-pass filter on the linear acceleration to reduce the effect of the noisy measurements with the following equation:

\[
s(i) = s(i - 1) + \delta(r(i) - s(i - 1)), i > 0
\]

Where s(i) is the \(i\)th smoothed linear acceleration signals, r(i) is the \(i\)th raw linear acceleration sample, and \(\delta\) is the smoothing factor. A smoothing factor of 0.25 gave us the best performance in our experiments.

3.5 Obtaining the User Motion Direction

To obtain the user motion direction, we transform the linear acceleration \(\ell\), which is the acceleration due to all forces applied to the phone except gravity, from the cell phone coordinate system to the user coordinate system to obtain \(\ell_u\). This is achieved by applying the quaternions to the linear acceleration vector \(\ell\):

\[
\ell_u = q \ell q^\ast
\]

where: \(q^\ast\) is the conjugate of the quaternion \(q\).

Once the linear acceleration is transformed to the user coordinate system, what remains is to obtain the user direction relative to North. Figure 9 shows that even though the phone has been oriented to North by applying the rotation angles, the linear acceleration samples are not aligned with North, but rather with the user direction. Humaine exploits this observation to detect the user direction as the direction that has the maximum variance. To obtain this direction, we consider only the two horizontal components of the transformed linear acceleration \((\ell_u = (\ell_F, \ell_S))\) that lie in the (F,S) plane, which is the user plane of motion. We then apply PCA on a window that contains values of \((\ell_F, \ell_S)\) components of the transformed linear acceleration \((\ell_u)\). The first principal component is the direction of maximum variance. Figure 10 shows the acceleration signal at different stages of processing by Humaine. We found that a window size of 3 seconds balances accuracy and latency.

3.6 Resolving Ambiguity

The obtained direction through PCA cannot differentiate between the actual user direction and its opposite, i.e., \(\theta_u\) and \(\theta_u + 180\). To resolve this ambiguity, we use the phone orientation angle \(\gamma\) as a hint and choose the PCA output direction that is closest to \(\gamma\). This will work as long as the difference between the y-axis of the phone and the forward direction of the user is within \(\pm 90\)°, which is the typical case as we see in Section 4.

4. Evaluation

We implemented Humaine on different Android devices including Samsung Galaxy Nexus S, Samsung Galaxy Tab 10.1, and an Asus Nexus 7 tablet. Raw sensor measurements were obtained through the Android API with sampling rate 50 Hz. To evaluate the system applicability under different magnetic field characteristics, we used 17 testbeds including...
different rooms at a typical home environment, a college library, office rooms, garden, college campus, different wide and narrow streets, among others. A total of 170 experiments with more than 7000 user direction estimates were performed by 10 users while placing the phones at different positions including pants pocket, shirt pocket, in bag, and in hand; as well as covering different phone orientations relative to the user direction of motion. To obtain ground truth, we marked the user path on the ground and used the path orientation from Google Maps as the ground-truth.

For the rest of the section, we compare the performance of the end-to-end system, in typical indoor and outdoor testbeds under different phone positions, with the Android API (as the current widely deployable system), uDirect system [13, 14] (closest related work), and GPS. The metrics used are absolute angle estimation error, latency, and power consumption.

4.1 Comparison Systems

In this section, we compare the performance of Humaine, in terms of heading estimation error, latency in direction change estimation, and power consumption; to uDirect [13, 14], the Android API, and GPS. uDirect uses inertial sensors available on the cell phone to estimate the user direction. However, it employs a step detection technique to detect the user heading. uDirect requires a new analytical model of the generated acceleration pattern for each new position [13, 14]. However, the only model presented and evaluated in details is for handling the acceleration pattern of the femur bone (i.e. in pants pocket position), and deriving models for other positions is not straightforward as we show later in this section. On the other hand, the Android API is the default heading estimation technique available on Android phones and has been used by a number of systems, e.g. [7, 26], to determine the phone orientation, rather than the human orientation (assuming both are the same). However, it is considered the widely deployed state-of-the-art. We also compare with the user bearing provided by the cell phone’s GPS as widely used outdoor technique that can estimate the user heading and does not depend on the phone position.

We start by evaluating the accuracy of the different techniques on a single segment to quantify the effect of different phone positions. We then perform a continuous trace experiment to study the latency of the different techniques and their suitability for real-time applications.

4.2 One Segment User’s Heading Estimation Error

Figures 11 and 12 show the CDFs of the user orientation estimation error for the three systems—Humaine, uDirect [13, 14], and the Android API—indoors and outdoors respectively for the different cell phone positions under 17 different testbeds. The experiments were performed by 10 users while placing the phones at different positions covering different phone orientations relative to the user direction of motion. We also show the GPS for outdoor experiments. Tables 3 and 4 summarize the results. Through this evaluation, we used straight segments, i.e. without including any changes in direction. We evaluate the effect of the change in direction on the techniques estimation error and latency in Section 4.3.

The results show that Humaine gives the best accuracy in all cases. Although both Humaine and uDirect use the magnetometer sensor, which is sensitive to magnetic noise in the environment especially indoors, we believe that Humaine is less sensitive due to two reasons: First Humaine depends on estimating the heading direction based on the variance of the user linear acceleration, which is less sensitive to noise compared to the rotated acceleration pattern within a step used by uDirect. Second, Humaine sensor fusion module described in Section 3.3 lessens the effect of the

![Figure 11: CDF plots comparing Humaine, uDirect [13, 14], and Android API indoors. Humaine outperforms other techniques in all positions. uDirect fails indoors, especially when the phone is not in the pants pocket it was designed for.](image1)

![Figure 12: CDF plots comparing Humaine, GPS, uDirect [13, 14], and Android API outdoors. Humaine outperforms all systems in all positions. uDirect fails when the phone is not in the pants pocket (the position it was designed for). GPS gives good accuracy outdoors. However, it has significant latency as we quantify in Section 4.3.](image2)
Table 3: Comparison between Humaine [13, 14], uDirect [13, 14], and Android API in indoor environments. Note that GPS does not work indoors. Percentages degradation are calculated relative to Humaine.

<table>
<thead>
<tr>
<th>Technique</th>
<th>Pants pocket</th>
<th>User Heading Estimation Error Indoors (degrees)</th>
<th>Hand held</th>
<th>Shirt pocket</th>
<th>Bag</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>90%</td>
<td>70%</td>
<td>Max.</td>
<td>90%</td>
<td>70%</td>
</tr>
<tr>
<td>Humaine</td>
<td>21</td>
<td>14</td>
<td>31</td>
<td>12</td>
<td>19</td>
</tr>
<tr>
<td>uDirect</td>
<td>50</td>
<td>32</td>
<td>70</td>
<td>12</td>
<td>19</td>
</tr>
<tr>
<td>Android API</td>
<td>50</td>
<td>17</td>
<td>50</td>
<td>22</td>
<td>37</td>
</tr>
</tbody>
</table>

Table 4: Comparison between Humaine, uDirect [13, 14], Android API, and GPS in outdoor environments. Percentages degradation are calculated relative to Humaine. GPS results do not depend on the phone position relative to user body.

<table>
<thead>
<tr>
<th>Technique</th>
<th>Pants pocket</th>
<th>User Heading Estimation Error Outdoors (degrees)</th>
<th>Hand held</th>
<th>Shirt pocket</th>
<th>Bag</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>90%</td>
<td>70%</td>
<td>Max.</td>
<td>90%</td>
<td>70%</td>
</tr>
<tr>
<td>Humaine</td>
<td>24</td>
<td>30</td>
<td>55</td>
<td>12</td>
<td>18</td>
</tr>
<tr>
<td>uDirect</td>
<td>24</td>
<td>40</td>
<td>90</td>
<td>12</td>
<td>18</td>
</tr>
<tr>
<td>Android API</td>
<td>55</td>
<td>60</td>
<td>180</td>
<td>40</td>
<td>49</td>
</tr>
<tr>
<td>GPS</td>
<td>30</td>
<td>57</td>
<td>177</td>
<td>31</td>
<td>57</td>
</tr>
</tbody>
</table>

4.3 Performance in a Continuous Trace

Figures 14 and 15 show the trajectory of the experiments to examine the effect of changing the direction during a continuous trace on Humaine and uDirect [13, 14] indoors and on Humaine, uDirect [13, 14] and GPS outdoors. Table 5 summarizes the results. For both cases, the phone was placed in the pants pocket as uDirect does not perform well in other cases, i.e. this is the best case scenario for uDirect as shown in the previous section. The results confirm that Humaine can smoothly track the user heading in both indoors and outdoors environments with most of the errors at the instances of direction change. Humaine also has the minimum latency, highlighting its suitability for applications that can tolerate delay. In addition, uDirect accuracy is significantly affected in indoor environments due to the noise affecting the acceleration pattern used in heading estimation (Figure 13).

GPS suffers from degradation in accuracy at areas without clear view of the sky, e.g. indoors, and urban canyons [2].

In addition, GPS has an initial fix latency with an average over different devices of 2 minutes. Moreover, it suffers from latency when changing direction with an average of 70 seconds. This latency affects the performance of GPS as shown in the experiment in Figure 15.
Table 5: Summary for the continuous-trace experiments. Phone placed only in pants pocket as uDirect completely fails in other positions as in Figure 11. Percentages degradation are calculated relative to Humaine.

The table shows that Humaine outperforms uDirect algorithm in both accuracy and latency with comparable energy-consumption.

4.4 Power Consumption

Figure 16 and Table 5 summarize the power consumption of the different algorithms. To calculate the power consumption, we used the PowerTutor profiler [27]. The figure shows that as expected, GPS has the worst energy consumption. uDirect involves searching for a specific pattern inside a step. Therefore, its complexity is linear in the step length. The results show that Humaine energy consumption is comparable to uDirect. This can be further enhanced by other techniques, e.g., duty-cycling.

5. CONCLUSION

We introduced Humaine, a system for robustly estimating the orientation of users carrying cell phones, suitable for new emerging crowd-sensing applications where the user has the freedom to have her phone in any arbitrary position or orientation. It fuses the phone’s energy-efficient inertial sensors and applies the PCA technique effectively to detect the user direction.

Evaluation of Humaine on Android devices with 170 experiments on different testbeds and phone positions shows that the system works accurately indoors and outdoors, without user involvement or prior configurations. Humaine achieved a median accuracy of 15°, which is 558% better than the state-of-the-art. Furthermore, Humaine has a minimal effect on the phone battery and latency, highlighting its suitability for real-time applications.

Currently, we are extending the system in different directions including employing other sensors available on the cell phone, e.g., the camera, opportunistically for a more accurate direction estimation, duty-cycling the sensors for...
further energy efficiency, dynamically adapting the system parameters, among others.

Acknowledgment
This work was supported in part by a Google Research Award.

6. REFERENCES