

ASSESSING DISABILITY COMPLIANCE USING SMARTPHONE

Mohammad Tanviruzzaman¹, Rizwana Rizia¹, Sheikh Iqbal Ahamed¹, and Roger Smith²

¹Marquette University, Milwaukee, WI 53233, ²University of Milwaukee-Wisconsin, Milwaukee, WI 53201-0413

ABSTRACT

Existing Americans with Disabilities Act Compliance Assessment Toolkits provide simple, pass-fail assessments of the requirements of the Americans with Disabilities Act (ADA) and other accessibility standards to assess the architectural barriers of the built environment. The toolkits include a Measurement Kit, which consists of various measuring instruments like, measuring tape, sound-level meter, light-level meter, magic slope block, etc. In this paper, we show that many of the instruments in the Measurement Kit can be effectively (having acceptable accuracy) implemented as a software application on a smartphone equipped with sensors like, accelerometer, gyroscope, and microphone which we call Americans With Disabilities Act Measurement Tool (ADAMT). We provide detailed algorithms and experimental results showing that an iPhone or iPod Touch 4.0 application (ADAMT) can serve as measuring tape, sound-level meter, as well as slope meter, making the ADA Compliance assessment both cheaper and more portable.

INTRODUCTION

Over 50 million Americans have some kind of physical, sensory, cognitive, or mental disability [1]. The Americans With Disabilities Act (ADA) of 1990 sets the minimum requirements (minimum door width, staircase slope, etc.) for state and local government facilities, public accommodations, and commercial facilities to be readily accessible to and usable by individuals with disabilities with a view to eliminate architectural barriers for these people. The most popular The Americans with Disabilities Act - Compliance Assessment Toolkits is ADACAT [11-12]. It is a screening tool-kit, which allows individuals without advanced training to assess the architectural

barriers of the built environment [1, 9]. It consists of two parts: Audit and Measurement Kit. Each audit is essentially a web-based questionnaire concentrating on the legal requirements of the ADA. The answers to the questions in an audit come from the actual measurements performed using the Measurement Kit. After answering the questions in an audit, the person, who is performing the accessibility-assessment, can click the "Score Audit" button to receive a numerical score of the degree of accessibility and usability compliance of say, a room [10].

The Measurement Kit consists of a measuring tape, a sound-level meter, a light-level meter, a door force tool, etc. In this paper, our focus is limited to implementing the measuring tape, the magic slope block, and the sound-level meter of the Measurement Kit as a smartphone application (ADAMT). The measuring tape is used to measure width of hallways and doors, height of signages, barriers, and clearances under desks and tables. A digital sound-level meter enables pass-fail assessments of the environmental sound-level with sufficient precision. The magic slope block enables one to measure slopes of ramps and paths, and cross-slopes of both [11].

In this paper we show that a smartphone equipped with accelerometer, gyroscope, and microphone can adequately (having acceptable accuracy) serve as a measuring tape, sound-level meter, and slope meter. In order to measure length or height (measuring tape) the accelerometer of the smartphone is used to perform dead reckoning based inertial navigation [2]. The basic idea is to integrate acceleration twice to get the position of the phone and the distance (length or height) is the difference between the final and the initial positions of the smartphone. The Microelectromechanicalsystem (MEMS) type accelerometer used in a smartphone is

susceptible to various deterministic and stochastic errors. Two consecutive integrations at each time instant cause any small error to grow large very quickly [3].

The deterministic errors include, bias, scale-factor, nonorthogonality, etc., whereas the stochastic errors include, random walk, Gauss-Markov process, etc [4]. We apply the 6-position static test to determine the principal deterministic errors (bias, scale-factor, and non-orthogonality), and then we use this information to calibrate the sensor-data, before applying stochastic error removal techniques [5]. In order to reduce the stochastic errors, we use Allan Variance technique for determining velocity random walk error coefficients of the accelerometer (data) and then we feed those coefficients into a Kalman filter along with the calibrated data [3, 6].

The smartphone's microphone is used to record audio data, and then the environmental sound-level is determined from the Fast Fourier Transform (FFT) coefficients calculated from the recorded audio data utilizing Parseval's identity to avoid the calculation of inverse FFT [7]. We use gravity and gyroscope data to measure slope.

MOTIVATION

At present, all the measuring instruments of ADACAT's Measurement Kit are accommodated in a 30" x 8" x 4" bag that weighs about 4 lbs. and it costs about \$650 [11]. Obviously, smartphone software capable of performing most of the measurements required by the ADA with acceptable accuracy can make the Measurement Kit cheaper (since many people already possess smartphones) and more portable; this is the principal motivation behind our work.

A database containing accessibility ratings of various parts of all the buildings can assist the people with disabilities to live independently. For example, with such a database in place, a person with disabilities can inquire about a building before actually going there or he may find some facility, e.g., a nearby restaurant, which has good accessibility scores. In order to populate such a database, the participation (may be voluntary) of a great number of individuals is needed. The goal is very unrealistic with the present Measurement

Kit of the ADACAT, which is too cumbersome and costly. On the other hand, if a person with disabilities feels that part of a building does not comply with the ADA standards, he can perform the measurements using his smartphone with much more ease than using the various instruments of the Measurement Kit of the ADACAT; and if needed he can then report his findings to the appropriate authority.

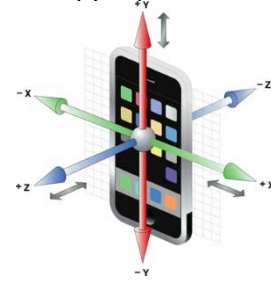


Figure 1: Smartphone's sensor axes

MEASUREMENT KIT ON SMARTPHONE

Measurement of length/height

The basic idea for measuring length and height using a smartphone is to apply dead reckoning based inertial navigation. The phone is moved through air from the starting point of the distance to be measured, to the ending point. The acceleration data received from the phone's accelerometer is integrated twice at each time instant to determine the current position of the phone [2]. We assume the starting point of the smartphone to have coordinates $\mathbf{0}^{3 \times 1}$ and thus the Euclidean distance between the last position, $[s_{x,last}, s_{y,last}, s_{z,last}]^T$ and the starting point is the distance of interest. So we have,

$$\text{distance} = \sqrt{s_{x,last}^2 + s_{y,last}^2 + s_{z,last}^2} \quad (1)$$

In order to determine the phone's position at each time instant we apply Newton's laws of motion along each of the three axes:

$$v_{axis,k} = v_{axis,k-1} + a_{axis,k} \delta t \quad (2)$$

$$s_{axis,k} = s_{axis,k-1} + v_{axis,k} \delta t + \frac{1}{2} a_{axis,k} \delta t^2 \quad (3)$$

Where, $k = 1, 2, \dots$ are time instants at which we determine the smartphone's positions; $v_{axis,k}$ is the speed of the phone at time instant k along an axis (x , y , or z) as depicted in Fig. 1; $a_{axis,k}$ is phone's acceleration received from the phone's accelerometer; δt is the time-difference

between two consecutive time-instants; and $S_{axis,k}$ is the phone's position. We describe the calibration of accelerometer data (to remove deterministic errors like: scale factor, bias, and nonorthogonality) in the experimental section. In order to reduce uncertainty of the calculated position of the smartphone due to random errors in the accelerometer data, we apply the Discrete-time Kalman Filter (DKF) [2].

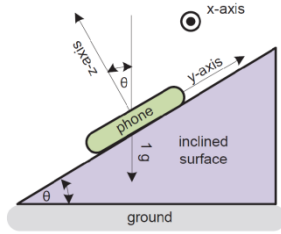


Figure 2: Slope using gravity

Measurement of Slope

Slope is the angle between the horizontal surface and a ramp or road. We determine the slope of an inclined surface in the following two methods:

1. Using gravity:

The phone is placed on the inclined surface in such a position that the slope of the surface is the angle of a counterclockwise rotation (θ) of the phone about its x-axis as shown in Fig. 2. Here $g = 9.8 \text{ ms}^{-2}$. Gravity along z-axis is $g_z = -\cos\theta$. So the slope is, $\theta = -\cos^{-1}g_z$. Core Motion Framework (CMF) of iOS 4.0 gives the value of g_z [12].

2. Using gyroscope:

The phone remains as in Fig. 2 and the roll (rotation about x-axis) value received from the gyroscope using CMF, provides the slope (ψ) directly.

Finally, we take the weighted average of the two slopes acquired from the above-mentioned two methods, as the slope (φ) of the surface [7]:

$$\varphi = \frac{w_1^2 \times \theta + w_2^2 \times \psi}{w_1^2 + w_2^2} \quad (4)$$

Where $w_1 = \sigma_1^{-1}$, $w_2 = \sigma_2^{-1}$, and σ_1 and σ_2 are standard deviations of θ and ψ .

Measurement of Sound-level

Human ear does not respond to all sound frequencies with equal sensitivity. As a result, even if two different noises have the same sound pressure level, any one of them could appear louder to a person when the sound power of that noise is concentrated in a frequency region where human ear is more sensitive. For this reason, frequency-weighting is used in sound level meters so that the meter becomes less sensitive at frequencies where human ear is less sensitive [8]. Among different frequency-weighting filters, A-weighting is the most popular and widely used one [13]. ADACAT's digital sound-level meter also employs A-weighting for measuring sound-level [14].

In order to measure sound-level, sound-data is recorded over δt length of time using the smartphone's microphone at a periodic interval of Δt ($\Delta t > \delta t$), then Fast Fourier Transform (FFT) is applied on the recorded sound-data. A-weighting is applied on the FFT coefficients and utilizing Parseval's identity we calculate mean energy of the recorded sound-data from the A-weighted FFT coefficients. Using Parseval's identity lets us avoid the calculation of inverse FFT, saving computational cost [7]. Finally the mean energy thus calculated is translated into decibel unit. In order to match the output of ADACAT's digital sound-level meter, we find a mathematical relationship between the smartphone's output and that of the digital meter, which we describe in the experiment section.

EXPERIMENT AND EVALUATION

Measurement of length/height

Most of the Americans With Disabilities Act (ADA)'s length or height measurements are limited to 5 feet [1]. In this section we show that an iPhone or iPod Touch 4.0 application can determine length or height of 5 feet or less with good accuracy given that both calibration and Discrete-time Kalman Filter (DKF) are used. For calibration we performed the 6-position static test [5]. In order to compute the process-noise covariance matrix and the measurement-noise covariance matrix to be used in DKF, we collected 6-hrs of static acceleration data at 100 Hz sampling

frequency, keeping the z-axis of the device pointing upwards on a level surface.

In order to compute the value of the coefficient of Velocity Random Walk (used in DKF process noise covariance matrix), we computed Allan variance for each axis of the accelerometer using the 6-hrs data (after subtraction of mean) [3].

Following table compares our distance measurement component with available iPhone applications for measuring a distance of up to 70.25":

App name	Method	Absolute error
AccelaRuler	Slide phone: injurious for phone	3.6"
Point & Measure	Need to know phone's height	4.5"
Sonic ruler	Need quiet room, smooth surface	Large
Our Algorithm	Move in air, easy, phone's safe	0.97"

Measurement of slope

We kept the iPod Touch flat on a horizontal surface keeping the z-axis pointing upwards, and collected a set of values for θ and ψ (as in Eq. 4) at a sampling rate of 0.5 Hz.

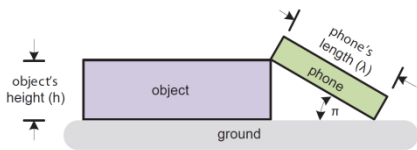


Figure 3: Slope using scale

From these values we calculated σ_1 ($\approx 1.40 \times 10^{-3}$) and σ_2 ($\approx 4.92 \times 10^{-4}$). To measure the reference values of various slopes, we kept the phone as in Fig. 3, varying the object's height (h). With a scale we measured the phone's (iPod Touch's) length λ and the object's height (h) and then the reference value of slope, $\pi = \sin^{-1} \frac{h}{\lambda}$, which is equivalent to using the magic slope block of ADACAT. Following table shows that measured slopes have sub-degree level precision:

Scale(π)	Phone(φ)	$ \pi - \varphi $
12.46°	12.40°	0.06°
27.88°	27.53°	0.35°
43.67°	43.42°	0.25°
53.67°	53.87°	0.20°
62.74°	62.94°	0.20°

Measurement of sound-level

We measured sound-levels at different places using both ADACAT's digital meter and

our algorithm. FFT was calculated using vDSP library of iOS 4 [15]. Fig. 4 shows the least-squares fit between two data sets:

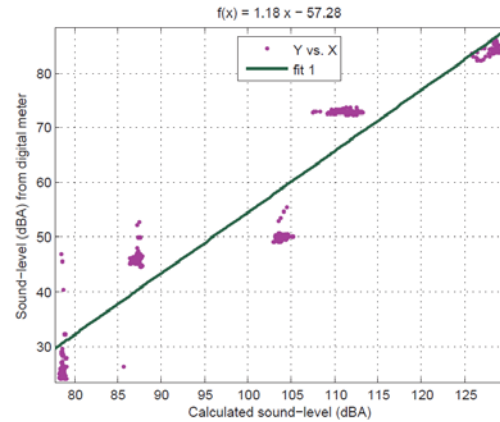


Figure 4: Fitted sound-level data

Later we used the fitted equation (on top of Fig. 4) to calculate sound-level in real time. The following table compares our algorithm with available iPhone applications for measuring sound-level:

App name	error
dB	9 dB
SPL meter	6 dBA
db meter	4 dB
Our	1.5 dBA

CONCLUSION

The experimental results show that it is feasible to implement existing toolkit's distance, slope, and sound-level measurement components as single smartphone software having acceptable accuracy that is more portable and less costly. Presently we are implementing two more components for measuring light-level and signage font's width-to-height ratio. These five components will cover most of the ADA measurement scenarios.

ACKNOWLEDGEMENTS

This work is supported in part by the Department of Education, National Institute on Disability and Rehabilitation Research (NIDRR), grant number H133G100211. The opinions contained in this paper do not necessarily represent the policy of the Department of Education, and you should not assume endorsement by the Federal Government.

REFERENCES

- [1] Rhoads, M. A. (2010). The ADA Companion Guide: Understanding the American's with Disabilities Act Accessibility Guidelines (ADAAG) and Architectural Barriers Act (ABA), John Wiley & Sons, Inc.
- [2] Titterton, D. and Weston, J. (2005). Strapdown Inertial Navigation Technology, American Institute of Aeronautics and Astronautics, 2nd edition.
- [3] Woodman, O. (2007). An Introduction to Inertial Navigation, Technical Report, UCAM-CL-TR-696, University of Cambridge.
- [4] Aggarwal, P., Syed, Z., Noureldin, A., and El-Sheimy, N. (2010). MEMS-Based Integrated Navigation, Artech House.
- [5] Aggarwal, P., Syed, Z., Niu, X., and El-Sheimy, N. (2007). A Standard Testing and Calibration Procedure for Low Cost MEMS Inertial Sensors and Units. In Journal of Navigation, vol. 61, no. 2, pp. 323-336.
- [6] IEEE Std. 952-1997 (1997). IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Interferometric Fibre Optic Gyros.
- [7] Strang, G. (2007). Computational Science and Engineering, Wellesley-Cambridge Press, 1st edition.
- [8] Harris, C. M. (1991). Handbook of Acoustical Measurements and Noise Control, McGraw-Hill, Inc., 3rd edition.
- [9] ADA Homepage. Retrieved, January 16, 2012, from <http://www.ada.gov>
- [10] ADA-CAT: making accessibility accessible. Retrieved on January 16, 2012 from <http://ada-cat.misericordia.edu>
- [11] ADA-CAT: Retrieved on January 16, 2012 from <http://www.aacoinstitute.org/ADA-CAT>
- [12] iOS Core Motion Framework reference. Retrieved on January 16, 2012 from http://developer.apple.com/library/ios/#documentation/CoreMotion/Reference/CoreMotion_Reference/_index.html
- [13] A-weighting filter. Retrieved on January 16, 2012 from <http://en.wikipedia.org/wiki/A-weighting>
- [14] Digital Sound-level Meter. Retrieved on January 16, 2012 from <http://www.mcmelectronics.com/product/TENMA-/72-935>
- [15] iOS vDSP library reference. Retrieved on January 16, 2012 from http://developer.apple.com/library/ios/#documentation/Accelerate/Reference/AccelerateFWRef/_index.html#//apple_ref/doc/uid/TP40009465