

Measuring Font Signage with a Smartphone Application for ADAAG Compliance Assessment

Farzana Rahman¹, Casey O'Brien¹, Colin Ostberg¹, Nahid Negar¹, Duc Do¹, Sheikh Iqbal Ahamed¹,
Roger O. Smith²

Marquette University, Milwaukee, WI 53233¹
University of Milwaukee-Wisconsin, Milwaukee, WI 53201-0413²

Abstract

It is very important to ensure that people with disabilities have a high quality of life in order to make their lives easier, perform daily tasks more comfortably and successfully. The existing Americans with Disabilities Act Compliance Assessment Toolkits (ADACAT) provide simple, pass-fail assessments of the requirements of the Americans with Disabilities Act (ADA). We propose a smartphone based application named Americans with Disabilities Act Measurement Tool (ADAMT) that will implement many of the tools of the ADACAT. Our focus in this paper is to present the design and algorithm of a new ADAMT component, "The Signage Font Meter".

Keywords

Americans with Disabilities Act; smartphone; font level; sound level; distance; compliance assessment; signage.

1. Introduction

In the United States, between 40 and 50 million people report some sort of disability [1]. The Americans with Disabilities Act (ADA) sets the minimum requirements for state and local government facilities as well as public and commercial facilities to be usable by individuals with disabilities in order to eliminate architectural barriers for these people [1, 2]. The ADACAT (Americans with Disabilities Act - Compliance Assessment Toolkit) provides simple pass-fail assessments of the ADA requirements [3, 4].

In this paper, we show that our smartphone based application can perform the functionalities of ADACAT. Our focus is limited to implementing the measuring tape, magic slope block, sound level meter and font guide tool of the ADACAT as a smartphone application. We present the Americans with Disabilities Act Measurement Tool (ADAMT), a smartphone based toolkit that has different applications performing the functionalities of the various tools of the ADACAT measurement kit. ADAMT has 4 different component applications performing the functionality of the measuring tape, magic slope block, sound level meter, and font guide. We refer to these applications as the distance meter, slope meter, sound meter, and font meter respectively. The detailed algorithms for the first three are presented in [5, 6]. The major focus of

this paper is to present a new component of ADAMT, the Signage Font Meter

1.1 Our major contributions

Our contributions in this paper are as follows:

- We propose the development of a smartphone based application named ADAMT. It can perform the functionalities of ADACAT measurement kit in a cheaper and more portable way than the typical mechanical tools. ADAMT application currently has 4 different components. The details of the distance meter, slope meter, and sound meter have been presented elsewhere [5, 6].
- This paper discusses the algorithm that performs the font signage's ratio measurement that uses only the smartphone's camera sensor.
- We also illustrate a screenshot of the prototype application of the font meter in this paper. The application was developed on an iPod touch. We evaluated it using the output of ADACAT's font guide as a reference value.

The rest of the paper is organized as follows: in Section 2 we give the motivation behind our work. We discuss the details of the ADAMT smartphone application in Section 3. The details and algorithm used in the font meter application is presented in this section as well as the prototype application. Finally, we conclude our paper with future research directions in Section 4.

2. Motivation

The ADACAT includes specially designed ADA measurement hand tools and carefully selected standard tools in a kit integrated with software for automatic checklist creation, making measurements, recording data, and much more. The ADACAT makes it easier to meet accessibility standards by simplifying the assessment of architectural barriers. However, all these tools are stored in a bag that weighs almost 4 lbs, and the combined tools cost about \$500 [7]. However, using only simple smartphone based applications the majority of the functionalities of the ADACAT can be performed with adequate accuracy and radically increased efficiency. The usage and access of such a smartphone based application targets the facilitation of disability compliance assessment. It ensures easy use, portability and lower purchase cost compared to the

ADACAT measurement kit. This is the principal motivation behind our work.

With a portable, cheap and less cumbersome tool such as the ADAMT smartphone application to verify disability compliance, building owners will have more incentive to achieve ADA compliance. Additionally, if a person with disabilities feels that part of a building does not comply with the ADA specifications, he/she can perform the measurements using a smartphone with much more ease than using the various instruments of ADACAT; and if needed he/she can then report his/her findings to the appropriate authority.

3. ADAMT: A Smartphone based Application

In this section, we provide an overview of 4 different components of the ADAMT smartphone based disability compliance assessment application. We discuss the detailed functionally and working processes of the signage font meter application.

3.1 Distance meter

One of the major components of the ADACAT is the measuring tape that is used to measure various types of distances such as the width of hallways and doors, barriers, and clearances under desks and tables. Our smartphone based application ADAMT allows such measurement with the help of the integrated sensors of the mobile phone such as the accelerometer. We refer to the component that performs distance measurement in ADAMT application as “*distance meter*”. Tanviruzzaman et al. implemented a smartphone based distance meter application that can provide a certain level of accuracy [5, 6]. The basic idea of the distance meter is very simple. The smartphone is moved through the air from the starting point of the distance to be measured, to the ending point. The change of the accelerometer data allows the application to calculate the distance. Finally, various techniques are used to increase the accuracy of the measurement [8, 9, and 10].

3.2 Slope meter

The magic slope block tool of ADACAT allows the measurement of the maximum slopes of ramps, paths, and cross slopes of both. It provides a simple, pass-fail test for all ADA mandated slopes. This tool also provides a test for the maximum gap between an elevator and the floor of a building. We refer to the slope measuring component of ADAMT smartphone based application as the “*slope meter*” and it can determine slope with adequate accuracy. The slope meter application is also developed by Tanviruzzaman et al. in [5, 6] and this can be used as a component in ADAMT. Using the gyroscope sensor of the smartphone and the angular difference between the orientations of the phone, then the application can compute whether the phone is on a horizontal plane or on a slope. This allows the slope meter to determine the slope of a path or ramp. The slope meter can also use various techniques [5, 6] to improve the accuracy of the measurement.

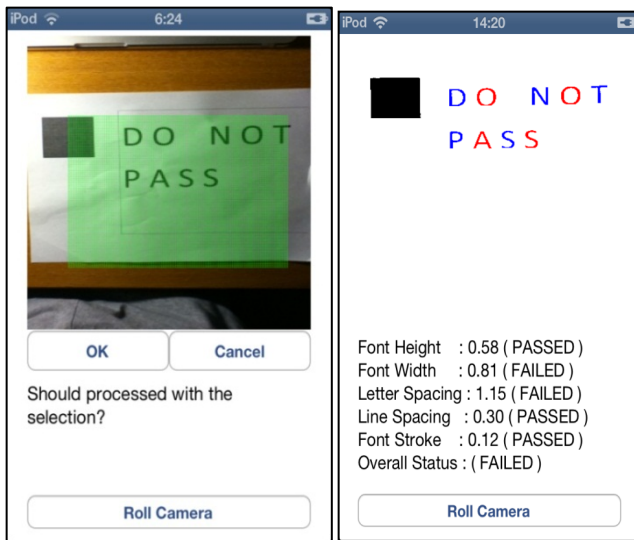
3.3 Sound meter

The sound level meter tool of ADACAT meets the needs of people with hearing limitations. Since the human ear responds with different sensitivities to sound with different frequencies, frequency weighting is used in the sound level meters of ADACAT so that the meter becomes less sensitive at frequencies where the human ear is less sensitive [11]. The sound meter of ADACAT has adequate precision and range in order to provide a simple pass-fail test of the environmental sound levels. We refer to the sound measuring component of ADAMT as the “*sound meter*”. The sound meter components of ADAMT are also developed and presented by Tanviruzzaman et al. in [5, 6] and can be used as a component in ADAMT. The smartphone’s microphone is used to record audio data and this is used by the sound meter application to determine the environmental sound level. In this application, “A-weighting” filter has been used to emphasize and suppress certain frequencies in the audio data so that the calculated sound level matches the response of a human ear [11].

3.4 Font meter

The “*Signage Font Guide*” tool of ADACAT is a very important component of the toolbox that allows, without much calculation, the ability to quickly determine that the a) sign font size, b) font width, c) line spacing, d) letter spacing, and e) stroke width do or do not meet the standard. The ADA standards for accessible signage are very complex, and they are described in terms of ratios that are based on the height of a letter. The data collection and formulas are sufficiently complex that many sign makers cannot reliably interpret them. The ADACAT manually operated Font Guide helps determine whether signage is accessible. The font guide is also one of the most complex tools of the ADACAT toolbox.

We refer to the font size measuring component of ADAMT as the “*Signage font meter*”. The user of the font meter first needs to prepare a single-colored square with known size to use as a “size reference”. Then the user needs to take a picture of the sign with letters and include the reference square in the picture. Then the application chooses the area containing the letters to start the analysis. There are three important steps (Step 1, Step 2 and Step 3) in this analysis phase that are shown in Figure 2, 3, and 4 respectively.



(a) taking picture of the font (b) displaying pass-fail test

Figure 1. Screenshot of the font meter application

STEP 1: Check if a reference square exists and whether it meets ADACAT Reference criteria

```

for each five color (black, white, red, blue, green) do
  1. Separate front ground and background by binarizing the
  image with Algorithm 1
  2. Group each pixel of the same type to a group with
  Algorithm 2
  3. Send the criteria of the reference box with Algorithm 3
  if a reference box is found do
    return reference box
  break
  end if
end for

```

Figure 2. Summary of workflow in Step 1 of font meter application

STEP 2: Identify the letters in the image

```

for each of five colors (black, white, red, blue, green) do
  1. Separate front ground and background by binarizing the
  font with Algorithm 1
  2. Group each font of the same type to a group with
  Algorithm 2
  3. Send the criteria of the font with Algorithm 3
  if one or more letters is found do
    return all the letters
  break
  end if
end for

```

Figure 3. Summary of workflow in Step 2 of font meter application

STEP 3: Check if the letter meets ADACAT requirements

1. Average font height width
2. Average line spacing
3. Average line stroke of all letters
4. Check if the above entities meet ADA specified font requirements

Figure 4. Summary of workflow in Step 3 of font meter application

• **Step 1:** In Step 1, the application checks whether the reference square meets ADACAT reference criteria. In this step of the font meter application three important algorithms (*Algorithm 1*, *Algorithm 2* and *Algorithm 3*) are executed sequentially for ADACAT criteria matching.

Algorithm 1 is referred to as “Apply binary filter with color”. The goal of this algorithm is to separate the front ground and background by making the image binary, i.e. binarizing the image. In this algorithm, we assume that the color of the reference box and the letter’s color are similar to one of five colors (black, white, red, and blue, green). Using this assumption, we binarize the image by filtering it through each of those 5 colors. *Algorithm 1* is shown in Figure 5.

Algorithm 2 is referred to as “Extract image objects”. The goal of this algorithm is to group each pixel of the same type to a group. This algorithm is shown in Figure 6.

Algorithm 3 is referred to as “Get single winning object”. The goal of this algorithm is to find the type of each pixel group. This algorithm is shown in Figure 7.

```

Algorithm 1: Apply binary filter with color
Output: a binarized image
for each of five colors (black, white, red, blue, green) do
  convert image to gray scale with respect to the color
  find minimum threshold and maximum threshold
  set each pixel in the foreground either to 1 or 0 using
  threshold to separate foreground and background
end for

```

Figure 5. Pseudo code for Algorithm 1

```

Algorithm 2: Extract image objects
Output: a pixel array
for each heightIndex do
  for each widthIndex do
    count pixelIndex
    if pixel is in foreground do
      if adjacent 8 pixel are in foreground do
        set all 9 pixels the same groupID
      end if
      count number of groupIDs
    end if
  end for
end for
return array of pixel array

```

Figure 6. Pseudo code for Algorithm 2

Algorithm 3: Get single winning object**Output: type of the object**

```

for each group of pixels that are both in foreground and have
same group ID
    get pixels in the corner
    get width and get height of the pixel
    get aspect ratio and aspect density of the image
    if the aspect ratio and aspect density of that group is on the
accepted range of concerning object type's thresholds
        set the group to be that type of object
    end if
end for

```

Figure 7. Pseudo code for Algorithm 3

- **Step 2:** If Step 1 is successful, the application moves to Step 2 where it identifies various letters in the image. This step also uses the three algorithms mentioned in Figure 6, 7, and 8 to identify various letters in the image. The workflow of Step 2 is shown in Figure 3.

- **Step 3:** The goal of step 3 is to check if the identified letters in step 3 meet the ADA specified requirements. This step performs simple average computation to measure the width and height of the fonts. Moreover, this step outputs and displays whether the font size measurements pass or fail the ADA specified rules. The workflow of this step is shown in Figure 4.

We have developed the font meter prototype application using an iPod touch. This application is a component of the ADAMT smartphone based application. The screenshots of the font meter are shown in Figure 1a and 1b.

4. Conclusions

In this paper, we described the idea of developing the ADAMT, a smartphone based application that can perform 4 functionalities of the ADACAT toolbox. The usage of this smartphone based application will allow disability compliance assessment to be more portable, easier, more efficient and at a lower purchase cost. We discussed the 3 major functionalities of ADACAT such as distance, slope and sound can be measured with adequate accuracy using a smartphone's accelerometer, gyroscope, and microphone. These 3 components are already developed and are presented in [5, 6]. This paper specifically detailed the Signage Font Meter. The design and algorithm to measure ADAAG compliance examines the ratio of height and width of the sign's font using a smartphone's camera. We implemented the algorithm on an iPod Touch and evaluated its performance using the output of ADACAT's font guide as a reference.

One important future research direction will be to further validate the measurement tools and develop additional tools of the ADACAT such as the light level meter, story stick, multitool, key force tool using a smartphone so that the ADAMT smartphone based

application can be used as simple substitute for the ADACAT toolbox. This will increase portability at a lower cost and will allow people with disabilities to have an easier means to check ADA specified regulations. Other possible future work will be to combine the audit and measurement components of ADACAT into a single smartphone application. In the future, we also intend to evaluate our ADAMT application on different smartphone platforms.

5. Acknowledgments

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.

6. References

- [1] M. A. Rhoads. The ADA Companion Guide: Understanding the Americans with Disabilities Act Accessibility Guidelines (ADAAG) and Architectural Barriers Act (ABA). John Wiley & Sons, Inc., 2010.
- [2] <http://www.ada.gov/>
- [3] <http://ada-cat.misericordia.edu/>
- [4] <http://www.aacoinstitute.org/ADA-CAT/>
- [5] M. Tanviruzzaman, R. Rizia, S. I. Ahamed, and R. Smith, "Assessing Disability Using Smartphone", in Proceedings of the Rehabilitation Engineering and Assistive Technology Society of North America (RESNA) Conference, 2012.
- [6] M. Tanviruzzaman, R. Rizia, S. I. Ahamed, and R. Smith, "Americans with Disabilities Act - Compliance Assessment Toolkit on Smartphone", in Proceedings of the IEEE COMPSAC 2012. pp. 442-451.
- [7] <http://www.aacoinstitute.org/ADA-CAT/>
- [8] P. Aggarwal, Z. Syed, X. Niu, and N. El-Sheimy. "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, 2007.
- [9] IEEE Std. 952-1997. "IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Interferometric Fibre Optic Gyros".
- [10] O. Woodman. "An Introduction to Inertial Navigation," Technical Report, UCAM-CL-TR-696, University of Cambridge, 2007
- [11] C. M. Harris. Handbook of Acoustical Measurements and Noise Control. McGraw-Hill, Inc., 3rd Edition, 1991.
- [12] Carin, L., Baraniuk, R.G. , Cevher, V. , Dunson, D. , Jordan, M.I. , Sapiro, G.; Wakin, M.B. , "Learning Low-Dimensional Signal Models," In IEEE Magazine of *Signal Processing*, vol.28, no.2. pp.39-51.
- [13] <http://developer.apple.com/library/ios/#documentation/CoreMotion/Reference/CoreMotionReference/index.html>
- [14] [http://developer.apple.com/library/ios/#documentation/Accelerate/Reference/AccelerateFWRef/index.html# //apple ref/doc/uid/TP40009465](http://developer.apple.com/library/ios/#documentation/Accelerate/Reference/AccelerateFWRef/index.html#//apple_ref/doc/uid/TP40009465)

