

# Evaluation of Smartphone-based Sound Level Meters

Trinity Cheng

River Hill High School, trinity.cheng@howardcc.edu

**Abstract** – Free, widely-available smartphone-based sound level meters have been utilized to collect large quantities of distributed data in short time periods for the efficient creation of crowd-sourced noise maps. However, the accuracy of these apps can vary greatly as previous studies have shown. In this study, four smartphone-based sound level meters were tested to evaluate their agreement. Four experiments were conducted to test the impact of different apps, operating systems, smartphone hardware, and microphones on app measurements at different sound levels. A combination of four apps, four smartphones, two operating systems, and two microphone types were used in the tests, as well as a hardware-based sound level meter. Errors were evaluated based on two evaluation methods—root mean square error and linearity. The experiment results show that all of the apps produced different readings with respect to the same input stimulus. In other words, each of the apps, operating systems, smartphone hardware, and external microphones influenced the accuracy of smartphone-based sound level meters. Due to the wide variation in measurements, the usage of uncalibrated smartphone-based sound level meters seems to be unacceptable for serious noise assessments. However, the high linearity displayed by some apps indicates the potential for increased accuracy through calibration by professional-grade instruments.

**Index Terms** – Acoustics, Sound Pressure Level, Sound Level Meter, Traffic Noise.

## INTRODUCTION

A sound level meter is a device used for acoustic measurements, usually providing a reading of the sound level in decibels. Most SLMs include the following components: microphone with preamplifier, amplifier, frequency weighting, input gain control, time averaging, and output indicator [1].

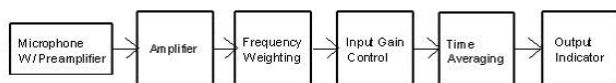


FIGURE I  
COMPONENTS OF AN SLM [1]

The microphone component is responsible for converting acoustic signals to electrical signals. Condenser

microphones, which can be further classified as either conventional and electret, are recommended due to their high stability, high sensitivity, excellent response at high frequencies, and low electrical noise characteristics [1]. A preamplifier provides high-input impedance and constant, low-noise amplification over a wide frequency range. The frequency weighting component takes into consideration the sensitivity of the human ear at different frequencies, producing readings that better reflect the human perception of loudness. A-weighting reflects the ear's response at lower sound pressure levels, B-weighting reflects the ear's response at moderate sound pressure levels, and C-weighting is essentially linear.

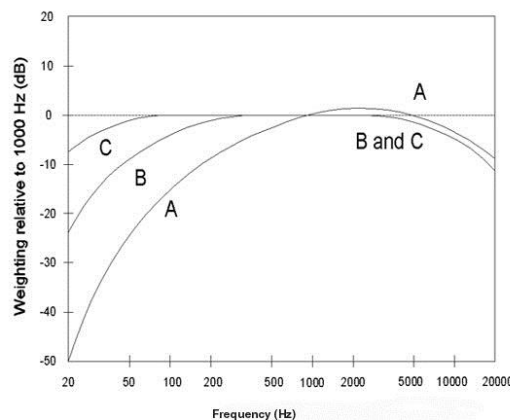


FIGURE II  
FREQUENCY WEIGHTING METHODS [1]

The time averaging component stabilizes the SLM's response to signals with changing amplitudes over time. Slow response is typically used for measurements of slowly-changing sound source levels, while fast response, which produces readings about eight times faster, is better for sound source levels which are changing more quickly.

The output indicator component is usually a display showing the sound level pressure in decibels (dB). For example, dB(A) indicates a sound level pressure reading measured with the A-frequency weighting.

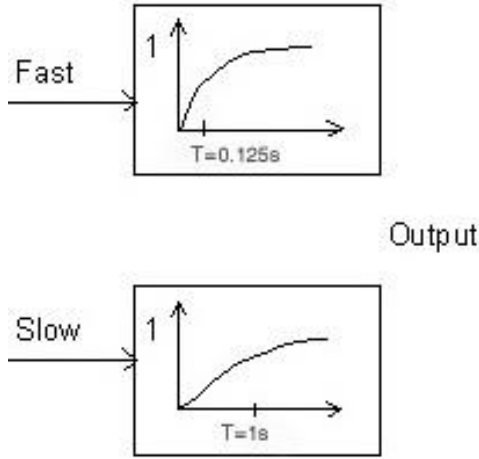


FIGURE III  
FAST AND SLOW TIME AVERAGING METHODS [1]

SLMs are certified as either Type 0, 1, or 2 based on their accuracy, with Type 0 having the highest precision [2]. Professional-grade Type 0 SLMs are expensive but offer high precision for laboratory measurements. In contrast, inexpensive, uncertified consumer-grade SLMs are also available with questionable accuracy.

Smartphone-based SLMs provide an alternative solution for measuring sound level at low cost. By taking advantage of the hardware components like the microphone, amplifier, and display that are readily available in billions of smartphones, smartphone apps only need to implement the software components of frequency weighting, gain control, and time weighting to provide sound level readings that can be immediately visualized, analyzed, and exchanged via the Internet.

Accurate smartphone-based sound level meters could allow for efficient creation of noise maps through crowd-sourcing [3]. Unfortunately, almost none of the smartphone apps available comply with international or national standards for sound level meters [2]. Previous studies have determined that the accuracy of these apps can vary greatly depending on factors such as hardware, software, noise source, and even phone age [3]. In order to produce accurate readings, the usage of external microphones and calibrators is recommended, but most non-expert users would not have the knowledge to access these instruments, resulting in inaccurate readings [4].

In the literature, a number of studies have investigated the accuracy of smartphone-based sound level meters. In 2014, a laboratory-based study compared smartphone apps with a Type 1 sound level meter in a reverberation room [3]. Certain iOS-based apps were found accurate and reliable based on their errors of less than 2 dB.

Another study tested five different apps on a single smartphone to isolate the variability in hardware [5]. Significant differences were found between the apps at most frequencies. The authors concluded that these smartphone-

based sound level meters are best used for entertainment rather than measurement purposes.

In another study conducted in 2016, 100 different smartphones running on either the iOS or Android platform were tested by measuring environmental noise levels. Based on the 1472 tests conducted, the authors concluded that lower standard deviation values suggest that iOS-based sound level meters are of superior performance and consistency than Android-based ones [6].

In 2018, a study conducted with professional-grade calibration showed that an app developed by the National Institute for Occupational Safety and Health running on an iOS smartphone with a professional-grade external microphone met most Class 2 requirements defined by international standards [2][4].

A smartphone-based sound level meter consists of components including the app, operating system, smartphone hardware, and microphone; each may affect its measurement accuracy. This study addresses the following questions: (1) Will different apps running on different smartphones report the same measurements? (2) Will the same app running on different operating systems report the same measurements? (3) Will the same app running on the same operating system but on different smartphones report the same measurements? (4) Will the same app running on the same operating system on the same smartphone using different microphones report the same measurements?

## METHODOLOGY

### I. MATERIALS

TABLE I shows the four smartphone-based sound level meters tested in this study. Two of them run on the iOS platform, and three on the Android. All of the apps reported A-weighted sound levels using the fast response setting. Screenshots of the sound level meters are shown in FIGURE IV.

TABLE I  
SMARTPHONE APPS USED IN THE STUDY

Code	App name	Manufacturer	OS
NIOSH	Sound Level Meter [7]	NIOSH	iOS
DXP	Decibel X Pro [8]	SkyPaw Co.	iOS/Android
SMT	Sound Meter [9]	Smart Tools Co.	Android
ABC	Sound Meter [10]	Abc Apps	Android

In this study, a hardware-based, consumer-grade sound level meter, DT-85A, was also included in all tests as a reference [11]. Similar to the smartphone-based sound level meters, DDT-85A reported A-weighted sound levels using the fast response setting. Notice that the findings in this study do not assume or depend on the accuracy of this meter.



FIGURE IV

SLMS TESTED IN THE STUDY. FROM LEFT: DT-85A, DXP/MOTO 4, NIOSH/iPHONE7, SMT/MOTO G4, AND ABC/S9+.

Four smartphones were used in this study as shown in TABLE II.

TABLE II  
SMARTPHONES USED IN THE STUDY

Code	Phone Model	Manufacturer	OS	Age
i7	iPhone 7	Apple	iOS 13.3.1	1
G4	Moto G4	Motorola	Android 7	3
X4	Moto X4	Motorola	Android 9	1.5
S9	Galaxy S9+	Samsung	Android 9	0.5

## II. Study design

Four test cases were conducted to answer the four research questions as shown in TABLE III.

TABLE III  
CASE CONFIGURATION

Case	Smartphone	App	Microphone	Frequency (kHz)
I	iPhone 7	NIOSH	Built-in	0.5, 1.0, and 1.5
	Moto X4	DXP		
	Moto G4	SMT		
	Galaxy S9+	ABC		
II	iPhone 7 Moto X4	DXP	Built-in	1.0
III	Moto G4 Moto X4 Galaxy s9+	SMT	Built-in	1.0
IV	Moto X4	DXP	Built-in External	1.0

In Case 1, four different apps running on four different smartphones were tested at three frequencies.

In Case 2, a single app, DXP, was tested running on either an iOS- or Android-based smartphone.

In Case 3, a single app, SMT, was tested running on the Android operating system on three different smartphones.

In Case 4, a single app, DXP, was tested running on a single Android-operated smartphone with and without an external microphone.

## III. Stimulus Generation

The sound stimulus was generated by the set-up depicted in FIGURE V. The idea behind this setup is to use digital input

as the independent variable to generate controlled sound levels and then examine whether the sound level meter readings correlate with the digital input. The correlation between the meter's reading and the ground truth is indeed the linearity property. Testing the linearity is useful when the absolute accuracy of the meter is not available. If the sound level meter is not absolutely correct but has perfect linearity, the previously collected measurement data can be corrected by adding a constant once a sound level meter calibrator is available.

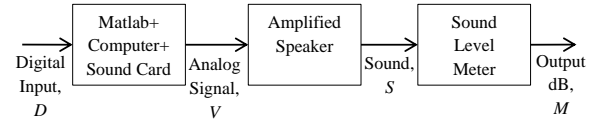


FIGURE V

SETUP GENERATING THE SOUND STIMULI

The Matlab programming environment was used to generate the sound stimuli. The digital input data, a sine waveform, was generated with amplitudes ranging from 0.1 to 0.5 at increments of 0.1 at frequencies of 0.5, 1.0, and 1.5 kHz. The waveform, defined in Equ 1, was played at a sampling rate of 8192 Hz by the Matlab sound() function, which drove the built-in sound card, essentially a digital-to-analog converter, in a Windows 7 desktop computer to generate analog audio signals,  $V$ , as defined in Equ 2. The analog audio signals were then amplified into audible sound waves,  $S$ , by a mini stereo system (Panasonic SB-PM03) with volume (i.e., gain) control fixed.

## IV. Measurement

The smartphones were secured 0.5 feet away from the face of the speakers with 0 degree incidence. Care was taken to ensure that vibration against the tabletop would be minimized. The appropriate sound level meter apps were opened so that the sound meters would begin measuring the sound pressure levels. As the audio signal played through the speakers, an iPad was used to take a picture of the device screens in order to ensure that the sound pressure levels were taken at the same time. In Case IV, the external microphone was secured so that its location and orientation were the same as the built-in microphone.

## V. Linearity

The digital input is a sine waveform written as

$$D = A \sin(2\pi f t), \quad (1)$$

where  $f$  is the frequency of the sound (e.g., 0.5, 1.0, and 1.5 kHz),  $t$  is time, and  $A$  is the amplitude (0.1, 0.2, ... 0.5), the independent variable in this experiment.

Assuming that the combination of Matlab, desktop computer and its sound card is a perfect digital-to-analog converter, the analog audio signal  $V$  in voltage is proportional to the digital input  $D$ .

$$V = kD = kA \sin(2\pi ft) \quad (2)$$

Assuming that the amplifier is perfectly linear at frequency  $f$  with a fixed impedance, the output sound power is proportional to the square of voltage according to Ohm's law

$$P = V^2/R, \quad (3)$$

where  $P$  is power,  $V$  is voltage, and  $R$  is resistance.

Power intensity  $I$  is the ratio of power to area

$$I = P/E, \quad (4)$$

where  $P$  is power and  $E$  is area.

The power intensity of sound is equal to the product of sound pressure level and particle velocity

$$I = SPL \cdot v, \quad (5)$$

where  $SPL$  is sound pressure level and  $v$  is particle velocity.

Thus we have

$$\frac{(kA \sin(2\pi ft))^2}{R \cdot E} = SPL \cdot v \quad (6)$$

When  $k$ ,  $R$ ,  $E$ ,  $v$  and  $f$  are constant,  $A^2$  is proportional to  $SPL$ .

$$A^2 \cong SPL \quad (7)$$

When  $A^2$  is represented in dB, the relationship between  $A$  and  $SPL$  becomes linear:

$$dB(A^2) = 20\log(A^2) = 40\log(A) \cong SPL \quad (8)$$

Therefore, the relationship between sound pressure level and amplitude in dB should be linear.

#### VI. Error evaluation

The errors were evaluated by root mean square error (RMSE) and linearity in this study. RMSE measures the difference between two outputs, while linearity quantifies the relationship between the input and the output.

RMSE between two measurements  $x$  and  $y$  is calculated by:

$$RMSE(x, y) = \sqrt{\sum_{i=1}^n \frac{(x_i - y_i)^2}{n}} \quad (9)$$

Linearity is represented by the coefficient of determination, which is calculated with the least squares of the line  $y = mx + b$ , which fits the data points.

## RESULTS

### I. Case I (different apps at different frequencies)

FIGURES V, VI, and VII show measurement results of the five sound level meters at 0.5, 1.0, and 1.5 kHz, respectively. The x-axis is digital amplitude, and the y-axis is measured sound pressure level (dB). Each meter has five data points corresponding to the five digital amplitudes.

It can be seen from the figures that all sound level meters reported different measurements in each test condition. The apps in order from lowest to highest sound pressure level readings at 0.5 kHz are ABC, SMT, NIOSH, DT-85A, and DXP. While NIOSH produced lower readings than DT-85A at 0.5 and 1.0 kHz, its readings became higher than DT-85A at 1.5 kHz. Higher frequency influences either NIOSH or DT-85A measurements.

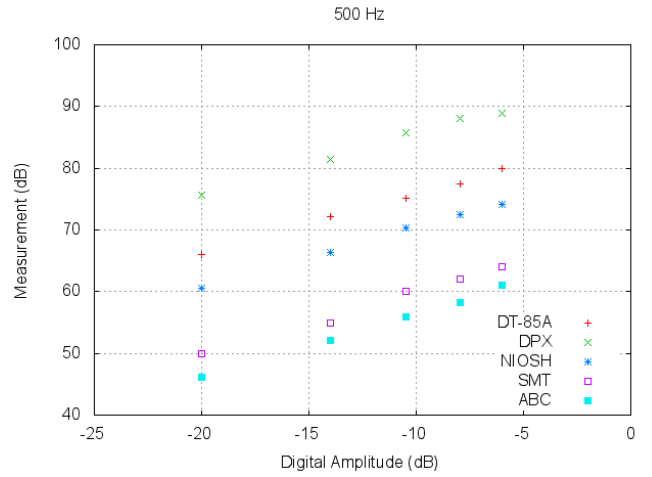


FIGURE VI  
MEASUREMENT DATA AT 0.5 KHZ

At 0.5 kHz, all apps displayed excellent linearity, with  $R^2$  values greater than 0.9900. This indicates that the apps can be calibrated by adding a constant in order to yield accurate results.

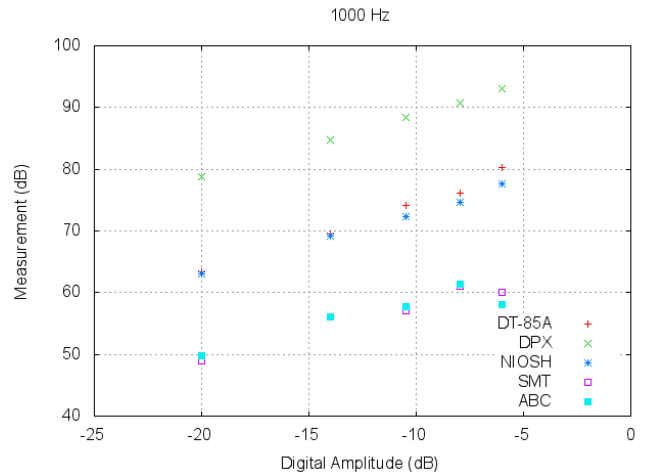


FIGURE VII  
MEASUREMENT DATA AT 1.0 KHZ

When the frequency was increased to 1.0 kHz, readings from DT-85A, DXP, and NIOSH remained linear, with  $R^2$  values of 0.9880, 0.9992, and 0.9959, respectively. However, ABC and SMT produced measurements that decreased as the digital amplitude increased near the maximum amplitude, resulting in decreased linearity.

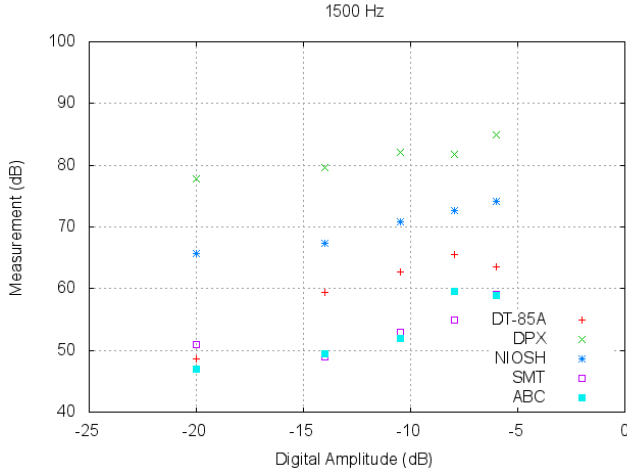


FIGURE VIII  
MEASUREMENT DATA AT 1.5 KHz

At 1.5 kHz, all of the apps as well as the DT-85A showed decreased linearity compared to at 0.5 kHz. There were multiple readings that stood out from the rest. At the minimum two amplitudes, SMT reported measurements in a reversed order. DT-85A, ABC, and DXP also exhibited similar abnormalities. The most significant drop in linearity was in SMT (difference of 0.3646) while the smallest difference was in NIOSH (0.0487).

The RMSE data between each pair of the five meters are shown in Table IV.

TABLE IV  
RMSE BETWEEN SLMs

SLM	DT-85A	DXP	NIOSH	SMT	ABC
DT-85A	-	16.10	7.02	14.01	15.31
DXP	16.10	-	14.19	28.16	29.34
NIOSH	7.02	14.19	-	14.32	15.36
SMT	14.01	28.16	14.32	-	2.66
ABC	15.31	29.34	15.36	2.66	-

ABC and SMT are the two meters with the smallest RMSE value (2.66), followed by NIOSH and DT-85A (7.02). The others have RMSE greater than 14.

Results of linearity are shown in TABLE V. All apps displayed excellent linearity at 0.5 kHz, with  $R^2$  values greater than 0.99. This indicates that the apps can be calibrated by adding a constant in order to yield accurate results. NIOSH seems to be best calibrated for different

frequencies due to the consistently high linearity. The combination of high amplitude and high frequency may have surpassed the other meters' capability, resulting in the inaccurate readings and decreased linearity.

TABLE V  
LINEARITY AS COEFFICIENT OF DETERMINATION OF DIFFERENT SLMs

Frequency (kHz)	DT-85A	DXP	NIOSH	SMT	ABC
0.5	0.9966	0.9922	0.9985	0.9913	0.9976
1.0	0.9880	0.9992	0.9959	0.9330	0.8268
1.5	0.8933	0.8911	0.9498	0.6267	0.8474

## II. Case II (same app, different OS)

FIGURE IX shows the measurement data of the DXP app running on iOS and Android.

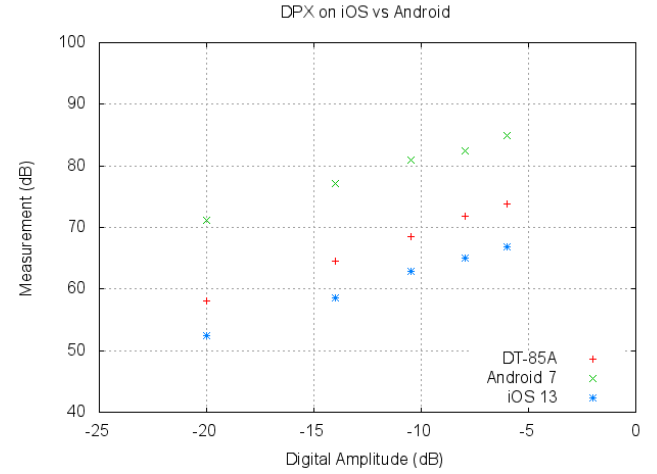


FIGURE IX  
MEASUREMENT DATA OF DXP ON DIFFERENT OPERATING SYSTEMS

The DXP app yielded very different readings on the two operating systems, with an RMSE of 18.17 dB. Smartphone X4 produced higher readings than the i7. Typically, as observed in Case I, DXP produced higher readings than DT-85A. However, DXP produced readings lower than DT-85A when running on an iOS-operated iPhone 7, indicating that a difference in operating systems may result in drastically different sound pressure level readings.

However, both smartphones displayed excellent linearity, with  $R^2$  values over 0.9900—X4 with 0.9968 and the i7 with 0.9977. This indicates that the DXP app can be calibrated on both iOS and Android devices in order to produce more accurate readings.

## III. Case III (same app, same OS, different phone)

FIGURE X shows the measurement data of the SMT app running on different Android smartphones.



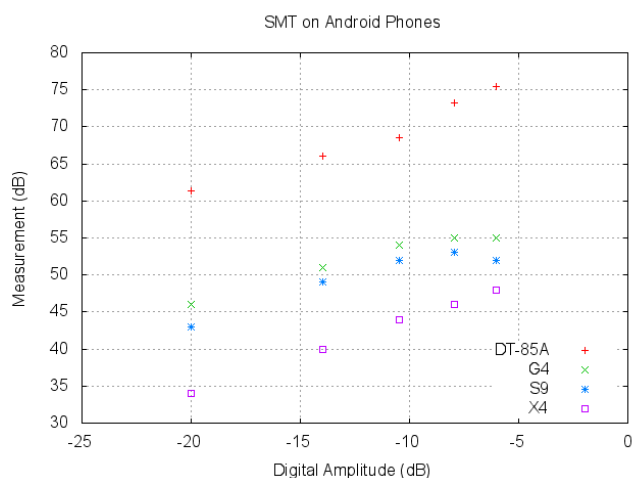


FIGURE X  
MEASUREMENT DATA OF SMT ON ANDROID PHONES

The readings of the same app running on different Android smartphones are different. Measurements on G4 and S9 were close with an RMSE of 2.45 dB compared with 7.63 dB (S9 and X4) and 9.95 dB (X4 and G4).

X4 produced the lowest readings of the three phones, but had the best linearity, with an  $R^2$  value of 0.9986 compared to 0.8971 (S9) and 0.9569 (G4).

#### IV. Case IV (same app, same phone, same OS, different microphone)

FIGURE XI shows the measurement data of the DXP app running on the same Android smartphone with and without an external microphone.

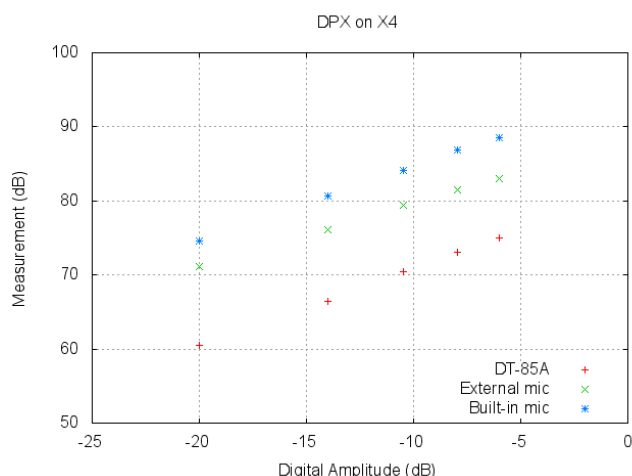


FIGURE XI  
MEASUREMENT DATA OF DPX WITH BUILT-IN VS. EXTERNAL MICROPHONE

Adding an external microphone resulted in higher measurements than those from the built-in microphone. The readings between the built-in and external microphones differed by an RMSE of 4.80 dB. Nevertheless, both the built-in and external microphones had very good linearity,

with  $R^2$  values greater than 0.9990. This indicates that the app can be calibrated by adding a constant to produce accurate readings. Additionally, SMT produced lower measurements than DT-85A in Case I, but produced higher measurements in Case IV for unknown reasons.

## CONCLUSION

The agreement of different smartphone-based sound level meters was investigated in this study. After testing in four configurations, the results show that the four apps produced different readings with respect to the same sound stimulus. In other words, each of the apps, operating systems, smartphone hardware, and external microphones influenced the accuracy of smartphone-based sound level meters. Due to wide variation in measurements, the usage of uncalibrated smartphone-based sound level meters seems to be unacceptable for accurate results. However, some of the apps displayed very good linearity, indicating the potential for accurate calibration by professional-grade instruments.

## FUTURE WORK

This study was conducted using household electronics so that the experiments could be easily reproduced by non-professionals. In future studies, a professional-grade sound level meter calibrator, which generates absolute sound levels, will be used as a reliable sound source to interrogate the smartphone-based sound level meters. The collected measurement data will be used to develop the calibration method.

## ACKNOWLEDGMENT

Many thanks to Ms. Susan Smith, Village Manager of River Hill, Columbia, Maryland, for her feedback and encouragement. Her call for residence participation in assessing the traffic noise level along the MD32 highway was the motivation for this study. I also thank my parents, who helped obtain publications and standards, and generously lent their phones for research in this study. Additionally, I would like to thank my brother for his patience.

## REFERENCES

- [1] U.S. Department of Transportation Federal Highway Administration, "Measurement of Highway-Related Noise," 1996. (Report No. FHWA-PD-96-046). [Online]. Available: <https://www.fhwa.dot.gov/environment/noise/measurement/mhnm03.cfm>
- [2] International Electrotechnical Commission. IEC 61672-1 (2nd ed.). Geneva, Switzerland: IEC, 2013.
- [3] Kardous, C. A., & Shaw, P. B., "Evaluation of smartphone sound measurement applications," The Journal of the Acoustical Society of America, 135(4), 2014, <https://doi.org/10.1121/1.4865269>. [Online]. Available: <https://asa.scitation.org/doi/10.1121/1.4865269>
- [4] Celestina, M., Hrovat, J., & Kardous, C. A., "Smartphone-based sound level measurement apps: Evaluation of compliance with international

- sound level meter standards,” *Applied Acoustics*, 139, 119-128, 2018, <https://doi.org/10.1016/j.apacoust.2018.04.011>.
- [5] Nast, D. R., Speer, W. S., & Le Prell, C. G, “Sound level measurements using smartphone "apps": Useful or inaccurate?” *Noise and Health*, 16(72), 251-256, 2014, <https://doi.org/10.4103/1463-1741.140495>.
- [6] Murphy, E., & King, E. A, “Testing the accuracy of smartphones and sound level meter applications for measuring environmental noise,” *Applied Acoustics*, 106, 16-22, 2016, <https://doi.org/10.1016/j.apacoust.2015.12.012>.
- [7] EA LAB. (2016). NIOSH Sound Level Meter (1.2.1). Accessed January 30, 2020. [Mobile application software]. Available: <https://apps.apple.com/us/app/niosh-sound-level-meter/id1096545820>
- [8] SkyPaw Co., Ltd. (2018). Decibel X Pro (4.4.3) Accessed January 30, 2020. [Mobile application software]. Available: [https://play.google.com/store/apps/details?id=com.skypaw.decibel10pro&hl=en\\_US](https://play.google.com/store/apps/details?id=com.skypaw.decibel10pro&hl=en_US)
- [9] Smart Tools Co. (2019). Sound Meter (1.7.2). Accessed January 30, 2020. [Mobile application software]. Available: [https://play.google.com/store/apps/details?id=kr.sira.sound&hl=en\\_US](https://play.google.com/store/apps/details?id=kr.sira.sound&hl=en_US)
- [10] Abc Apps. (2019). Sound Meter (3.5.1). Accessed January 30, 2020. [Mobile application software]. Available: [https://play.google.com/store/apps/details?id=com.gamebasic.decibel&hl=en\\_US](https://play.google.com/store/apps/details?id=com.gamebasic.decibel&hl=en_US)
- [11] CEM DT-85A Mini Sound Level Meter. (2012, May 8). [Online]. Available: <https://www.amazon.com/CEM-DT-85A-Sound-Level-Meter/dp/B0081FYMQ>

#### AUTHOR INFORMATION

**Trinity Cheng** is a 10<sup>th</sup> grade student at River Hill High School and Howard Community College, Maryland. She is interested in music, physics, and anything connecting the two subjects. She has placed first in state-level Science Olympiad Optics and Thermodynamics events. As a cellist, she has participated in many orchestras, including Maryland All-State, Baltimore Symphony Youth Orchestra, Howard County High School GT Orchestra, etc. She serves on the Teen Advisory Committee, Village of River Hill, Howard County, Maryland.