applied system innovation

MDPI

Article

On Smartphone Power Consumption in Acoustic Environment Monitoring Applications

Sergey Zhidkov *, Andrey Sychev, Alexander Zhidkov and Alexander Petrov

Cifrasoft Ltd., Izhevsk 426001, Russia; andrey.sychev@cifrasoft.com (A.S.); alex.zhidkov@cifrasoft.com (A.Z.); alexandr.petrov@cifrasoft.com (A.P.)

* Correspondence: sergey.zhidkov@cifrasoft.com; Tel.: +7-341-260-0526

Received: 18 January 2018; Accepted: 14 March 2018; Published: 19 March 2018

Abstract: In this paper, we provide a systematic analysis of smartphone power consumption in background audio recording scenarios. We report comprehensive measurement results for several Android smartphone models with diverse hardware specifications, running different types of Android operating systems (OS). Our experimental results show that with respect to background audio sensing applications, the major impact on smartphone power consumption is associated with the core audio recording functionality. On the other hand, moderately complex digital signal processing algorithms (e.g., sound pressure level estimation, acoustic feature extraction, and audio fingerprinting) that operate simultaneously with background audio recording do not significantly affect overall smartphone power consumption or battery life. Power consumption varies significantly from model to model, and we found no obvious correlation between the power consumption of background audio recording scenarios and factors such as the year of release, the OS version, or the model of smartphone processor or system-on-chip (SoC) hardware.

Keywords: smartphones; power consumption; acoustic monitoring; digital signal processing

1. Introduction

Recently, there has been growing interest in the use of inexpensive off-the-shelf smartphones in various innovative applications beyond regular voice calls. A modern smartphone is equipped with multiple sensors, including a microphone, which is perhaps the most ubiquitous and universally available sensor to third-party application developers. Several emerging mobile applications in such fields as healthcare and environmental monitoring require mobile devices that continuously perform microphone sensing and real-time digital signal processing in background mode. Examples of such applications include:

- Mobile applications that measure a person's exposure to background acoustic noises to assess the risk of noise-induced hearing loss (also known as "noise dosimeters") [1,2];
- Applications for crowd-sourced noise pollution monitoring and urban noise mapping [3–5];
- Sleep quality monitoring [6], estimation of respiratory rates [7], and obstructive sleep apnea monitoring applications [8];
- Smartphone-based TV and radio audience research applications [9,10];
- In-door location and proximity detection based on near-ultrasound acoustic beacons [11,12].

One of the major challenges in developing practical applications that require background audio monitoring is the power consumption of the microphone sensor and the associated digital signal processing. The Android operating system (OS), which is one of the most widely used OSs in the smartphone market, allows audio recording in background mode. However, the commercial feasibility of the abovementioned applications strongly depends on the impact of background audio sensing on a smartphone's battery life [13].

A number of previous studies have been devoted to the analysis of smartphone power consumption in different use scenarios [14–18]. However, most of these works do not explicitly analyze the background audio recording mode. For example, in [14] the authors analyze the power consumption of Wi-Fi use, the graphical processing unit, and video recording of four mobile devices (Samsung Galaxy line). In [18], the authors report the results of power consumption measurements for typical multimedia applications that extensively use a smartphone display. Finally, in [19], the authors report the power consumption of the microphone recording mode, yet they only provide the results for the Samsung Galaxy S6 smartphone.

In this paper, we provide a systematic analysis of smartphone power consumption in background audio recording scenarios. We report comprehensive measurement results for several Android smartphones with diverse hardware and software specifications. We also report the power consumption results for typical real-time digital signal processing algorithms that are frequently used simultaneously with background audio recording.

2. Methodology

We selected 18 mid-range and low-end Android smartphones from various device manufacturers, running different versions of the Android OS. In our test sample, one device was running Android 7.0, three devices were running Android 6.x, five devices were running Android 5.x, eight devices were running Android 4.x, and one device was running Android 2.3. These devices and their essential characteristics are listed in Table 1.

To perform power consumption measurements, a test smartphone was connected to the digital programmable power supply Rigol DP832A with the high-resolution option [20], and the digital programmable power supply was linked to a notebook PC via USB interface for loggingsubsequent data analysis. The schematic diagram of the experiment and the test setup are illustrated in Figure 1.

To automate the process of power consumption measurements in different modes of operation, we developed a custom Android application that periodically performed microphone audio recording and digital signal processing using several signal processing algorithms with different levels of complexity. The audio signal from the microphone was captured using the Android *AudioRecord* class [21] at a sampling rate of 44.1 kHz, and all digital signal processing algorithms were implemented in optimized fixed-point C++ code and integrated into the test application via the Android Native Development Kit [22].

To minimize the influence of other hardware and software components on the measurement results, we turned off 2G/3G/LTE mobile data transfer, disabled Wi-Fi and Bluetooth radios, and uninstalled or disabled all non-system applications (social networks, messengers, etc.) that could consume battery power in background mode.

Firstly, we ran audio recording tests without any signal processing at all. In this mode of operation, the audio samples that were received in the audio recording callbacks were simply ignored and the processing thread was inactive. Secondly, we run three real-time digital signal processing algorithms: "simple" (algorithm #1), "moderately complex" (algorithm #2), and "fairly complex" (algorithm #3), listed in Table 2. Each algorithm was active for 15 min with a 2 min "cool down" interval between runs, and then the measurement cycle was repeated multiple times. Each experiment lasted for at least 24 h. Instant (second-by-second) power consumption measurements were recorded using a digital programmable power supply and transferred to the notebook PC via USB interface. A typical sequence of recorded power consumption measurements is illustrated in Figure 2. As one can see, the instant power consumption never drops below a certain minimum level (in our example, 170 mW), and has occasional spikes as high as 650 mW (algorithm #3). These power variations are caused by the complex and dynamic nature of Android OS behavior. Moreover, all modern Android smartphones use dynamic central processing unit (CPU) frequency scaling (CPU throttling). Due to computational complexity, algorithm #3 usually causes the CPU governor to switch to a higher CPU frequency, which results in an overall power consumption increase (see Figure 2).

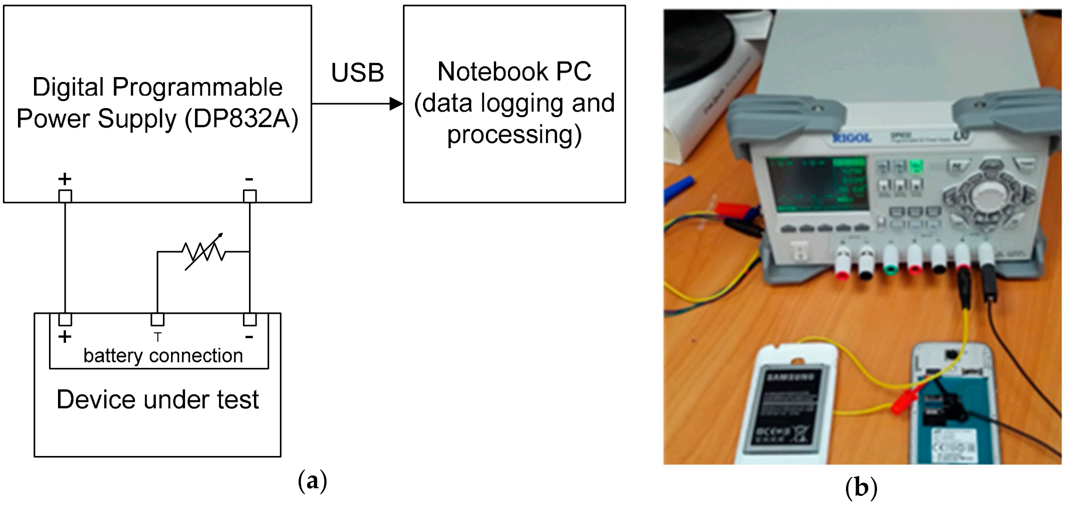

Digital Programmable Power Supply (DP832A)
+
-
USB
Notebook PC (data logging and processing)
+
T
-
battery connection
Device under test
(a)
(b)

Figure 1. (**a**) A typical schematic diagram of the test environment; and (**b**) an example of test setup.

Table 1. The list of tested Android smartphones and their essential characteristics.

Smartphone Model	OS	SoC/Processor	CPU Core	Battery
Lenovo A1000	5.0	SC7731	4×1.3 GHz Cortex-A7	2000 mAh
Fly FS510	6.0	MT6580	4×1.3 GHz Cortex-A7	4000 mAh
Samsung Galaxy S4	4.2	Exynos 5410	4×1.2 GHz Cortex-A7	2600 mAh
Prestigio PAP4055Duo	4.1	MT6577T	2×1.2 GHz Cortex-A9	2500 mAh
LG G2 mini	5.0	MSM8226	4×1.2 GHz Cortex-A7	2440 mAh
LG Optimus L3	2.3	MSM7225A	800 MHz Cortex-A5	1500 mAh
Alcatel Pixi 4 (5)	6.0	MT6735M	4×1.0 GHz Cortex-A53	2000 mAh
LG L50	4.4	MT6572	2×1.3 GHz Cortex-A7	1900 mAh
Micromax Juice (Q3551)	6.0	SC8830	4×1.2 GHz Cortex-A7	3000 mAh
Samsung Ace 2	4.1	DB8500	2×800 MHz Cortex-A9	1500 mAh
Samsung Galaxy S2	4.1	Exynos 4210	2×1.2 GHz Cortex-A9	1650 mAh
Meizu M3	5.1	MT6750	8×1.5 GHz Cortex-A53	2870 mAh
Samsung Galaxy S4 mini	4.2	MSM8930AB	2×1.7 GHz Cortex-A7	1900 mAh
Lenovo S890	4.1	MT6577	2×1.2 GHz Cortex-A9	2250 mAh
Philips S386	7.0	MT6580	4×1.3 GHz Cortex-A7	5000 mAh
Samsung Galaxy Tab 2	4.1	OMAP4430	2×1.2 GHz Cortex-A9	4000 mAh
Leagoo Z1	5.1	MT6580	4×1.3 GHz Cortex-A7	1300 mAh
ZTE Blade L5	5.1	MT6572	2×1.3 GHz Cortex-A7	2150 mAh

Table 2. Experimental algorithms used for the power consumption evaluation.

Algorithm No.	Description	Algorithm Complexity Measure [1]
Algorithm #1	A-weighting, sound pressure level estimation, simple audio feature extraction in time-domain [3,23]	0.9
Algorithm #2	Algorithm #1 + calculation of audio fingerprints (audio hashes) [24]	7
Algorithm #3	Detection and decoding of near-ultrasound communication signal (greedy synchronization algorithm, OFDM demodulator, Viterbi decoder)	88

[1] Algorithm complexity was measured in millions of required CPU cycles per second on a single core digital signal processing platform TMS320C6748 clocked at 300 MHz [25].

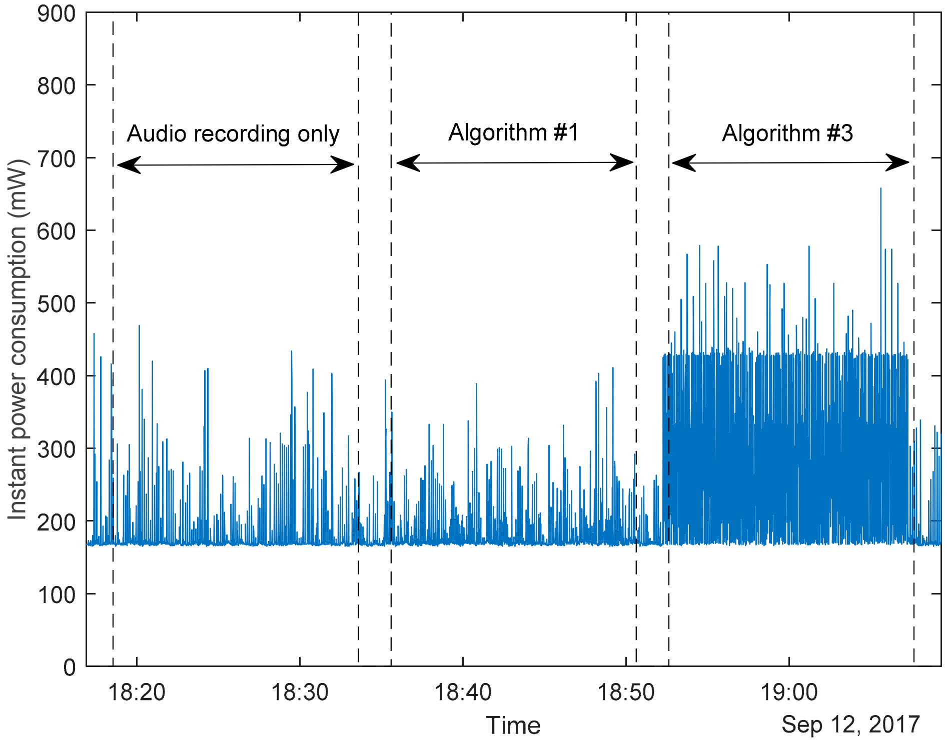


Figure 2. Example of instant power consumption measurements using the developed test application.

3. Results and Discussion

3.1. Power Consumption in Background Audio Recording with and without Digital Signal Processing

The second-by-second power consumption readings for all signal processing algorithms were averaged and plotted in Figure 3, along with 99% confidence intervals. As one can see, the difference between the devices with the lowest and highest power consumption may be quite high (up to six times in our experiments). It is also interesting to note that "simple" and "moderately complex" digital signal processing algorithms do not significantly contribute to overall smartphone power consumption. Only the most complex signal processing algorithm #3 visibly increases smartphone power consumption. Nonetheless, even the extreme optimization of algorithm #3 (for example, down to complexity of algorithm #1) would reduce the power consumption by 28% on average, while in certain smartphones the potential power consumption reduction would be only 7%–15%.

Figure 4 shows the expected battery life for all 18 tested smartphones in the case of continuous audio recording without any signal processing and in the case of audio recording and simultaneous real-time digital signal processing (algorithm #3). The median expected battery life was approximately 28 h with algorithm #3 and approximately 40 h without any digital signal processing. Nonetheless, 3 out of 18 devices (17%) were not able to operate longer than 22 h in continuous audio recording mode even without any real-time digital signal processing. It is worth mentioning that these results represent a rather idealistic scenario, because we disabled mobile data transfer, Wi-Fi and Bluetooth radios, all third-party background applications, and used the nominal battery capacity to predict battery life. In a realistic device usage scenario, the battery life would be shorter due to the battery degradation effect [26].

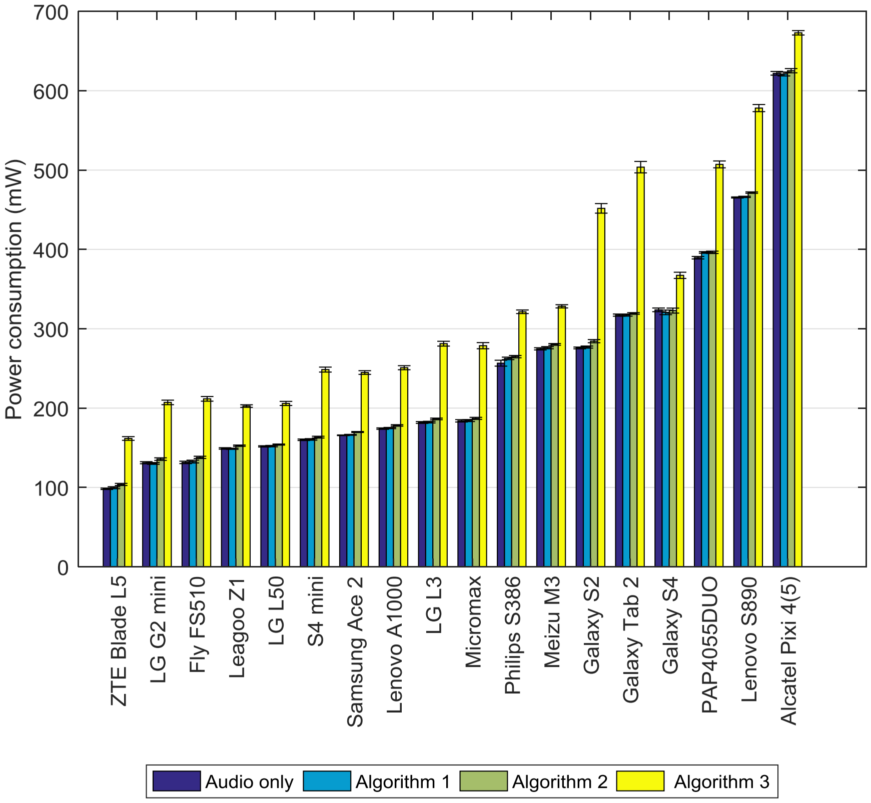


Figure 3. Average power consumption with confidence intervals for the 18 tested Android smartphones in four different modes of operation.

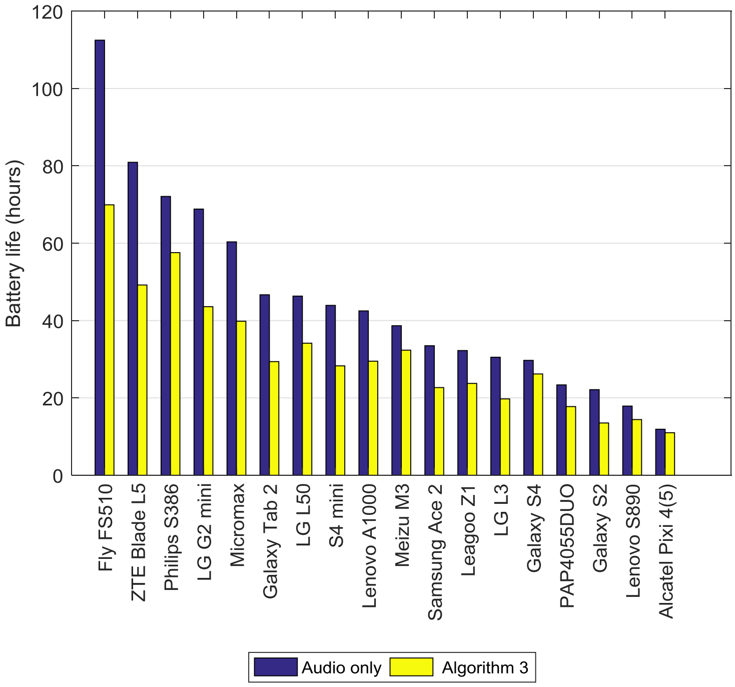


Figure 4. Expected battery life of the fully charged smartphones in audio recording only mode and "Algorithm 3" mode.

3.2. Impact of Audio Sampling Rate

The results presented in the previous subsection were obtained when the audio signal was sampled at 44.1 kHz rate. To evaluate how the audio sampling rate affects the power consumption, we performed a series of experiments at other sampling rates, namely 8 kHz, 22.05 kHz, and 48 kHz, using a subset of our test devices. For fair comparison, we only tested the audio recording mode. Based on the results of these experiments, we found no statistically significant evidence that smartphone power consumption depends on audio sampling rate.

3.3. Power Consumption in Wakelock Mode vs. Audio Recording Mode

The Android platform relies on several techniques to reduce overall mobile device power consumption. However, these techniques may cause undesirable side effects such as slowing down data processing or disabling certain hardware peripherals. To avoid these undesirable effects, the Android OS has a special software mechanism called *wakelock*, which prevents the device from going into a power-saving mode [27]. In fact, misuse of the wakelock mechanism is often regarded as the most common reason for the abnormal battery drain caused by mobile applications [28].

We used the same methodology described in Section 2 to measure the power consumption of our test devices in partial wakelock mode (CPU is active, but the screen and the keyboard are turned off [27]), and compared it with background audio recording power consumption. The measurement results for a few selected devices are illustrated in Figure 5. As one can see, the audio recording mode consumes up to 2×–6× times more power than the partial wakelock mode.

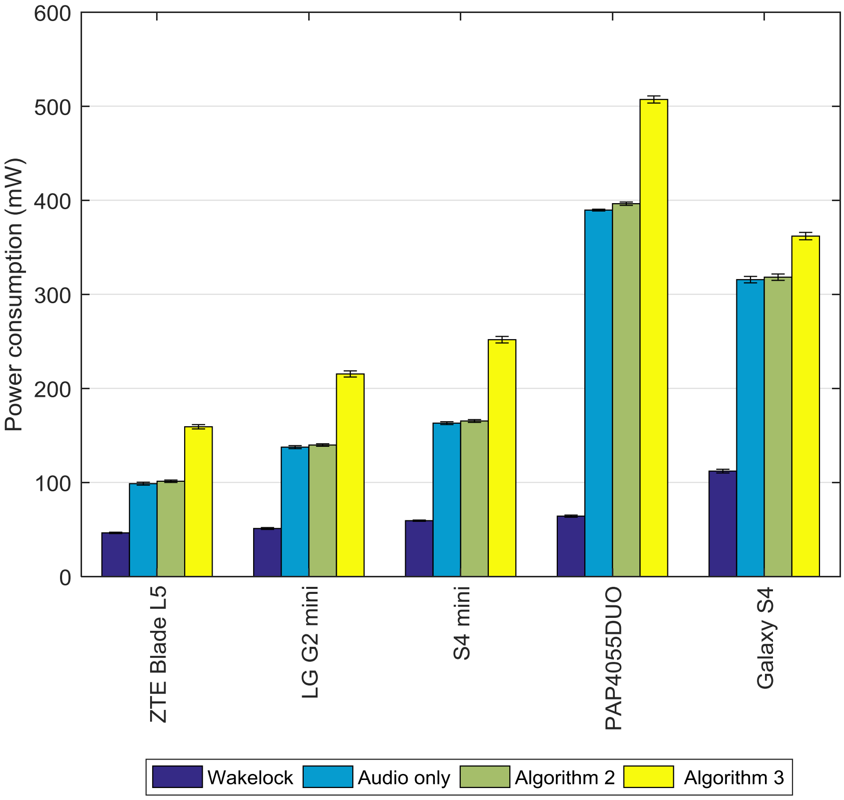
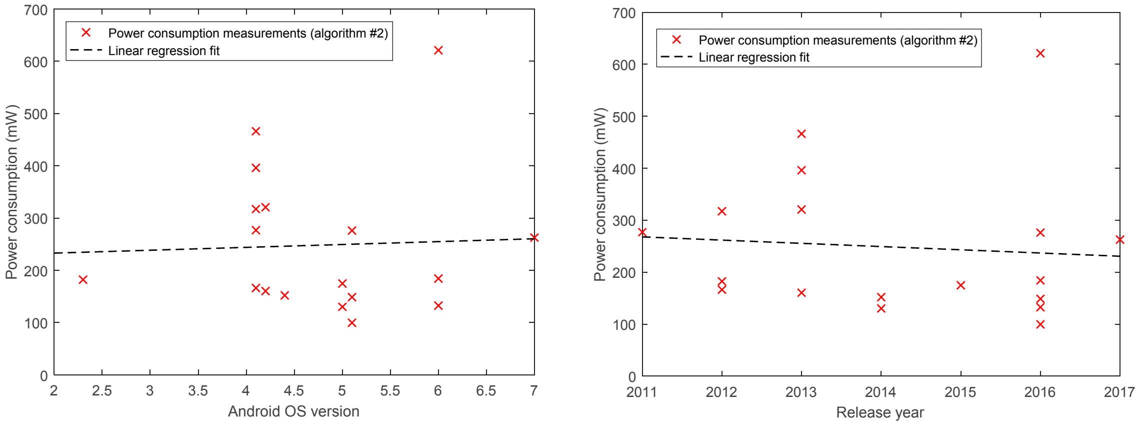


Figure 5. Average power consumption with confidence intervals for a subset of the tested Android smartphones (wakelock mode vs. audio recording and digital signal processing modes).

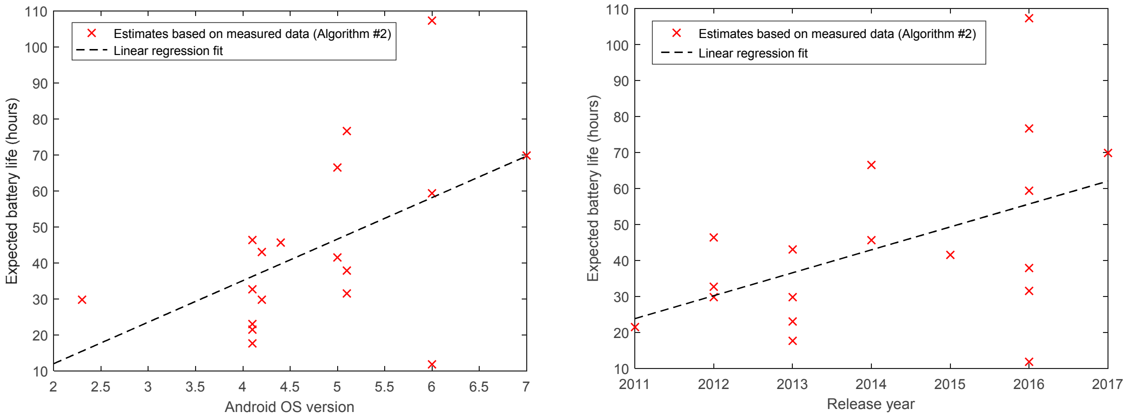
3.4. Power Consumption vs. Release Year and the OS Version

Since we discovered that the power consumption for different smartphone models may vary significantly from model to model (see Figure 1), we were interested to find out which hardware or software features may influence power consumption. Somewhat surprisingly, our results did not show any obvious correlation between the power consumption in background audio recording mode and hardware and software features such as the Android OS version, CPU and chipset model, the size of RAM and built-in memory, or the smartphone release date. For example, in Figure 6 we plot the measured consumption data vs. Android OS version number and the smartphone release date. One might expect that the newer smartphones would be more power efficient in audio recording mode, yet this is not the case. There is no statistically significant difference (at least, for our sample size) between power consumption of the smartphones released in 2011–2014 and the smartphones released in 2015–2017. On the other hand, if we consider the expected battery life in background audio recording mode vs. the smartphone release date, there is a clear upward trend (see Figure 7). However, this is easily explained by the fact that the newer smartphones tend to have larger capacity batteries than their older counterparts.



(**a**) (**b**)

Figure 6. (**a**) Power consumption vs. Android OS version and (**b**) the smartphone release year: measured data and the linear regression fit.



(**a**) (**b**)

Figure 7. (**a**) Expected battery life vs. Android OS version and (**b**) the smartphone release year: measured data and the linear regression fit.

4. Power Optimization Techniques

Our experimental results show that in background audio monitoring applications, the major impact on smartphone power consumption is caused by the core audio recording functionality. Therefore, in applications that perform continuous microphone sensing, optimization of digital audio signal processing algorithms often does not provide any noticeable battery-life savings.

The power-saving strategies that can be used in such scenarios are application specific. In most cases, the only viable power-saving approach is to perform the sound capture at certain moments (i.e., rely on non-continuous audio recording). For example, such an approach has been explored in [19], where the sound capture was triggered by events initially recognized by the accelerometer.

On the other hand, in TV audience measurement applications (see, for example, [9,10]), the audio recording can be performed periodically at short time intervals (e.g., 10–20 s) followed by longer intervals of microphone inactivity (e.g., 20–90 s). The interval of inactivity can be adaptively selected to match the TV watching profile of the individual user; this concept is schematically illustrated in Figure 8. Our experiments indicate that the TV audience measurement applications, which rely on such an adaptive mechanism, may reduce the average power consumption by a factor of 2–3 without significant impact on the accuracy of the measurement results.

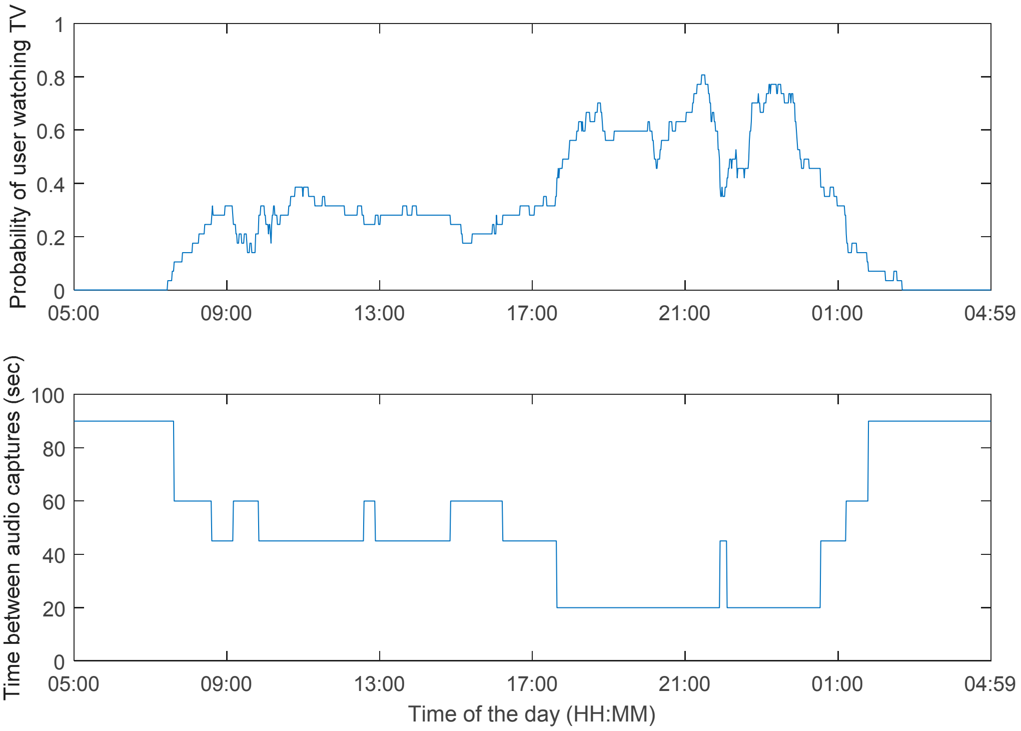


Figure 8. Adaptation of intervals between audio capturing events to the TV viewing profile of a particular user in the TV audience measurement application (real-world example).

The interval of inactivity may also be varied adaptively based on the audio features extracted during previous microphone sensing periods. For example, in noise exposure monitoring applications [1,3–5], the interval of microphone inactivity could be inversely proportional to the mean and variance of sound pressure level estimates obtained during previous sound capturing intervals. Nonetheless, it should be noted that in some applications, such as healthcare applications [6–8], it is vital to perform continuous audio recording and signal processing; in this case, the only practical solution is to use smartphone models with low or moderate power consumption in audio recording mode.

5. Limitations of the Study

The most obvious limitation of the study is its relatively small sample size: we analyzed the power consumption of only 18 Android devices. Although the selected subset of smartphones represents devices from different manufacturers and includes various types of Android OSs, it is nonetheless very small in comparison to the number of Android smartphone models available on the market (as of October 2017, the Google Play database of supported Android devices contained more than 15,000 models). Thus, some conclusions, especially those reached in Section 3.4, should be used with caution. In addition, we focused mainly on devices in the low-end and mid-range categories. Future research on the topic should include a wider selection of test devices, and should evaluate more high-end smartphone models.

6. Conclusions

Our analysis provides ample evidence that the major impact on smartphone power consumption of background audio monitoring applications is caused by the core audio recording functionality, whereas moderately complex audio processing algorithms (such as sound pressure level estimation, acoustic feature extraction, audio fingerprinting, etc.) that run simultaneously with background audio recording do not significantly impact overall smartphone power consumption and battery life. Therefore, in the majority of cases, the optimization of real-time digital audio signal processing algorithms may not provide any noticeable battery life savings if the mobile application runs in the background and performs continuous microphone sensing.

The power consumption caused by background audio recording varies significantly from model to model, and the difference between the best and worst smartphone may be significant. In most cases, the power consumption with audio recording enabled is much higher than the power consumption in wakelock mode.

Power consumption is likely influenced by multiple factors (both hardware-related and software-related), and we found no obvious correlation between power consumption in background audio recording mode and factors such as year of release, the OS version, and the CPU, processor, or system-on-chip model. However, batteries in newer smartphone models tend to last longer than those in older models, mostly because newer smartphones use larger-capacity batteries.

We believe that our results may be useful to developers of mobile applications that rely on background audio recording and real-time digital audio signal processing. The results may also be interesting to mobile device manufacturers.

Acknowledgments: This work was funded by Cifrasoft Ltd.

Author Contributions: Sergey Zhidkov designed the research, analyzed the data, and wrote the paper; Andrey Sychev co-designed the experiment, developed the test application, and performed the experiments; Alexander Zhidkov co-designed the experiment, developed the signal processing code, and prepared the test equipment; and Alexander Petrov performed the algorithm complexity measurements.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Roberts, B.; Kardous, C.; Neitzel, R. Improving the accuracy of smart devices to measure noise exposure. *J. Occup. Environ. Hyg.* **2016**, *13*, 840–846. [CrossRef] [PubMed]
2. Kardous, C.A.; Shaw, P.B. Evaluation of smartphone sound measurement applications. *J. Acoust. Soc. Am.* **2014**, *135*, EL186. [CrossRef] [PubMed]
3. Zamora, W.; Calafate, C.T.; Cano, J.C.; Manzoni, P. Accurate Ambient Noise Assessment Using Smartphones. *Sensors* **2017**, *17*, 917. [CrossRef] [PubMed]
4. Ibekwe, T.S.; Folorunsho, D.O.; Dahilo, E.A.; Gbujie, I.O.; Nwegbu, M.M.; Nwaorgu, O.G. Evaluation of mobile smartphones app as a screening tool for environmental noise monitoring. *J. Occup. Environ. Hyg.* **2016**, *13*, D31–D36. [CrossRef] [PubMed]

5. Aumond, P.; Lavandier, C.; Ribeiro, C.; Boix, E.G.; Kambona, K.; D'Hondt, E.; Delaitre, P. A study of the accuracy of mobile technology for measuring urban noise pollution in large scale participatory sensing campaigns. *Appl. Acoust.* **2017**, *117*, 219–226. [CrossRef]
6. Ong, A.A.; Gillespie, M.B. Overview of smartphone applications for sleep analysis. *World J. Otorhinolaryngol.-Head Neck Surg.* **2016**, *2*, 45–49. [CrossRef] [PubMed]
7. Nam, Y.; Reyes, B.A.; Chon, K.H. Estimation of Respiratory Rates Using the Built-in Microphone of a Smartphone or Headset. *IEEE J. Biomed. Health Inform.* **2016**, *20*, 1493–1501. [CrossRef] [PubMed]
8. Al-Mardini, M.; Aloul, F.; Sagahyroon, A.; Al-Husseini, L. Classifying obstructive sleep apnea using smartphones. *J. Biomed. Inform.* **2014**, *52*, 251–259. [CrossRef] [PubMed]
9. Bisio, I.; Delfino, A.; Luzzati, G.; Lavagetto, F.; Marchese, M.; Fra, C.; Valla, M. Opportunistic estimation of television audience through smartphones. In Proceedings of the 2012 International Symposium on Performance Evaluation of Computer & Telecommunication Systems (SPECTS), Genoa, Italy, 8–11 July 2012; pp. 1–5.
10. Ibrahim, M.; Gruteser, M.; Harras, K.A.; Youssef, M. Over-The-Air TV Detection Using Mobile Devices. In Proceedings of the 26th International Conference on Computer Communication and Networks (ICCCN), Vancouver, BC, Canada, 31 July–3 August 2017; pp. 1–9.
11. Liu, K.; Liu, X.; Li, X. Guoguo: Enabling Fine-Grained Smartphone Localization via Acoustic Anchors. *IEEE Trans. Mobile Comput.* **2016**, *15*, 1144–1156. [CrossRef]
12. Aguilera, T.; Paredes, J.A.; Álvarez, F.J.; Suárez, J.I.; Hernandez, A. Acoustic local positioning system using an iOS device. In Proceedings of the International Conference on Indoor Positioning and Indoor Navigation, Montbeliard-Belfort, France, 28–31 October 2013; pp. 1–8.
13. Greenemeier, L. People Love Their Smartphones but Hate the Batteries [Survey Results]. Scientific American, 28 November 2014. Available online: https://www.scientificamerican.com/article/people-love-their-smartphones-but-hate-the-batteries-survey-results/ (accessed on 7 March 2018).
14. Chen, X.; Nixon, K.W.; Chen, Y. Practical power consumption analysis with current smartphones. In Proceedings of the 29th IEEE International System-on-Chip Conference (SOCC), Seattle, WA, USA, 6–9 September 2016; pp. 333–337.
15. Bai, G.; Mou, H.; Hou, Y.; Lyu, Y.; Yang, W. Android Power Management and Analyses of Power Consumption in an Android Smartphone. In Proceedings of the IEEE 10th International Conference on High Performance Computing and Communications & IEEE International Conference on Embedded and Ubiquitous Computing, Zhangjiajie, China, 13–15 November 2013; pp. 2347–2353.
16. Datta, S.K.; Bonnet, C.; Nikaein, N. Android power management: Current and future trends. In Proceedings of the First IEEE Workshop on Enabling Technologies for Smartphone and Internet of Things (ETSIoT), Seoul, Korea, 18 June 2012; pp. 48–53.
17. Zhang, L.; Tiwana, B.; Qian, Z.; Wang, Z.; Dick, R.P.; Mao, Z.M.; Yang, L. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In Proceedings of the Eighth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis, Newport Beach, CA, USA, 1–3 October 2010; ACM: New York, NY, USA, 2010; pp. 105–114.
18. Chen, X.; Chen, Y.; Ma, Z.; Fernandes, F.C. How is energy consumed in smartphone display applications? In Proceedings of the 14th Workshop on Mobile Computing Systems and Applications, Jekyll Island, Georgia, 26–27 February 2013; ACM: New York, NY, USA, 2013; p. 3.
19. Lee, S.; Lee, J.; Lee, K. VehicleSense: A reliable sound-based transportation mode recognition system for smartphones. In Proceedings of the IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Macao, China, 12–15 June 2017; pp. 1–9.
20. Rigol DP832A Digital Programmable Linear DC Power Supply Datasheet and Specifications. Available online: https://www.rigolna.com/products/dc-power-loads/dp800/ (accessed on 17 December 2017).
21. Android Developers Portal: android.media.AudioRecord Class Documentation. Available online: https://developer.android.com/reference/android/media/AudioRecord.html (accessed on 17 December 2017).
22. Android Developers Portal: Android Native Development Kit (NDK) Documentation. Available online: https://developer.android.com/ndk/index.html (accessed on 17 December 2017).
23. Lu, L.; Zhang, H.J.; Jiang, H. Content analysis for audio classification and segmentation. *IEEE Trans. Speech Audio Process.* **2002**, *10*, 504–516. [CrossRef]

24. Haitsma, J.; Kalker, T. A Highly Robust Audio Fingerprinting System with an Efficient Search Strategy. *J. New Music Res.* **2003**, *32*, 211–221. [CrossRef]
25. TMS320C6748 Fixed- and Floating-Point DSP Datasheet. Available online: http://www.ti.com/product/TMS320C6748 (accessed on 17 January 2018).
26. Xu, B.; Oudalov, A.; Ulbig, A.; Andersson, G.; Kirschen, D. Modeling of Lithium-Ion Battery Degradation for Cell Life Assessment. *IEEE Trans. Smart Grid* **2018**, *9*, 1131–1140. [CrossRef]
27. Android Developers Portal: PowerManager Class Documentation. Available online: https://developer.android.com/reference/android/os/PowerManager.html (accessed on 17 December 2017).
28. Pathak, A.; Jindal, A.; Hu, Y.C.; Midkiff, S.P. What is keeping my phone awake? characterizing and detecting no-sleep energy bugs in smartphone apps. In Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys'12), Windermere, UK, 25–29 June 2012; ACM: New York, NY, USA, 2012; pp. 267–280.