Noise Sensing Calibration Under Different Phone Context

Min Huang(✉) and Lina Chen

South China University of Technology, Guangzhou, China
minh@scut.edu.cn

Abstract. Noise pollution severely threatens human well-being. Constructing a noise map based on crowd-sensing can help city planners better understand environmental noise at lower cost. Based on the strict sampling method limitation of state-of-the-art techniques, we build a new calibration model aimed at the pocket situations and the scenarios happened more frequently in actual life. The proposed model consists of a Activity Recognition Model (ARM) and a Signal Processing Model (SPM). Three types of data are taken into consideration, which are sitting, standing, and walking. In ARM, we collect 3-axis accelerometer data to identify current sampling context based on the convolutional neural network. SPM mainly implements noise level measurement and calibration according to the corresponding output of ARM under different phone context. The average errors after calibration are controlled to be within ±3 dB (A), and the classification precision reaches 99.2%. Finally, we display the noise map adopting different criteria based on the building types, which is more scientific and meaningful. The final results show that our proposed calibration model is feasible and can improve the data quality under different situations.

Keywords: Mobile sensors · Noise sensing · Phone context

1 Introduction

Urban noise pollution is increasingly serious along with the development of the industrial era and it becomes another unignored environmental problem following air pollution and water pollution. In 2017, Ministry of Ecology and Environment (MEE) of China received a total of 550,000 environmental noise complaints, accounting for 42.9% of the total environmental complaints [1]. Given that long-term disturbance of noise will greatly affect human health and raise the risk of tinnitus, heart disease and cardiovascular disease [2], establishing a noise map can effectively monitor urban sound level.

Building up a noise map is very expensive for the government to deploy dense sensor networks covering the whole city. Recently, thanks to the diverse and advanced sensors embedded in smartphones, crowd-sensing technology seems to be a promising solution by leveraging massive amount sensing data contributed by a large number of users [3] and provides opportunities for citizens to interact with the ambient environment [4, 5]. Most of today's smartphones are equipped with many powerful sensors, such as gyroscopes, accelerometers, microphones, proximity sensors, GPS, and digital

Y. Yin et al. (Eds.): MobiCASE 2019, LNICST 290, pp. 18–31, 2019.
https://doi.org/10.1007/978-3-030-28468-8_2

compasses, which makes mobile phones not only phones but also tools. Some papers have researched and implemented noise mapping based on this technology [6–8] and proved its feasibility and precision [9]. However, most of them require users to hold phones in their hands, which greatly limits the available conditions because only 6.27% people choose to hold phones in their hands instead of putting them in their bags or their pockets [10]. In this paper, we consider the pocket situation and use activity recognition techniques to study the effects of different behaviors on mobile phone measurements and how to correct them. Human activity recognition (HAR) can automatically detect human behavior and has been studied for a long time in many fields. Unlike conventional HAR, mobile phones play the role of sensing nodes in this paper, so only the equipped sensors can be utilized to identify users' behaviors and the 3-axis accelerometers are most commonly used to detect activity information.

We study the relationship between the measurements of mobile phones and the readings of sound level meters (SLM) in different phone context and propose a calibration model. This model can automatically adjust the corresponding sub-model according to the user's current activity to improve the data quality when the mobile phone is placed in the pocket. The proposed architecture consists of two models, Activity Recognition Model (ARM) and Signal Processing Model (SPM). ARM takes 3-axis accelerometer data as input and processes it by activity recognition technology based on convolutional neural network (CNN). ARM can identify three daily activities, walking, standing, and sitting by directly leveraging time series input without any manual operation. By adopting corresponding calculation sub-models in SPM, we can improve the accuracy of sensing data in previously unavailable situations, so we can get more data to analyze and construct a noise map.

The rest of this paper is organized as follows. In Sect. 2, we give an overview of relate work. In Sect. 3, the proposed noise sensing method is described in detail. In Sect. 4, we conduct the experiments and analyze the experimental results. Finally, Sect. 5 concludes this paper and discusses the future work in this field.

2 Related Work

As for noise sensing in intelligent terminals, microphones play a vital role in the system implementation. But the hardware differences between mobile phones and standard SLMs make it challenged to sense noise accurately. Mobile phones adopt Dual-Mic Noise Suppression technology to reduce the impact of background sound while SLMs equip a windshield to capture as much sound as possible [11–13]. Therefore, calibration is inevitable to make mobile phones work as SLMs. For improving data quality under different phone context, many researchers choose to use some extra devices. In [14], an additional microphone with foam windscreen was installed into the mobile phone so that no matter how the users moved the measurements reached relative stable and accurate results after avoiding the influence of wind and friction. The study [15] proved that using additional microphones installed into mobile phones could greatly improve measurements accuracy and suggested using them. However, It is obviously not so easy and convenient for everyone to get such an external microphone. So researchers try to introduce context-aware and HAR technologies into noise sensing.

In [16], they proposed a new architecture mainly including call detection module, signal processing module, speech detection module, and context discovery module. Sensing context was divided into hand and pocket or bag using a k-nearest neighbor algorithm, and the samples were adopted only when the phone was in the hand. In [17], a coarse-grained data analysis method was presented to classify collecting situations. The author made a set of criteria for evaluating every situation identified by accelerometer, gyroscope, proximity sensor and GPS. Each collection situation would be scored and data was filtered according to that. Only the qualified data would be adopted, which could effectively avoid the influence of friction and vibration caused by user movements.

In order to fully use any possible data, HAR technologies are non-trivial to identify and understand human performance. Differing from conventional methods requiring high computing power for data fusion and image processing, HAR based on mobile sensors embedded in smartphones must take low power and limited sensor types into consideration. The accelerometers can determine the phone direction and are most commonly used in HAR of mobile phones. WISDM Lab [18] collected six daily activities and classified them using traditional machine learning classification algorithms like J48, Logistic Regression, and Multilayer Perceptron. However, this kind of classification algorithms needs to manually extract features, which requires domain knowledge. Therefore, deep learning booms. The deep learning approaches can directly use sensing data as input and reduce much pre-work. CNN is the most frequent and effective-proven model of deep learning approaches used in HAR [19]. Once we solve the input adaption problem caused by the difference between our input time series data and images, using CNN in HAR will become relatively easy. In [12, 13], the 3-axis time series data was considered as the 3-channel input like RGB images and conducted the 1D convolution operation to them. However, the study in [20] argued that 1D kernel could not capture the dependency between different sensors in different positions. Therefore they present a CNN model with a 2D kernel to handle this. They concluded two types of modalities including sensors in different positions and different types. For the first model, they grouped the data according to positions, and for another model, they used padding zeros to avoid disturbance.

3 Noise Sensing Approach

3.1 The Proposed Architecture for Noise Sensing

In this paper, we propose a method for improving the data quality and for relieving data sparsity for noise evaluation and map construction. The hand-hold phone context is not the focus of this paper, so how to distinguish between hand hold and pocket context will not be discussed here, as there are already some papers solved this problem [21, 22]. What we try to do is to leverage sensing data even though the phones are put into the trouser pockets (front and right) in an upside-down way and that's the premise of our architecture. Because of that, the mobile phone is closely attached to the user's leg, and the current activity of the user can be predicted by using an accelerometer to sense the direction of the mobile phone and its changing law.

In this section, we describe the proposed architecture of the whole system in detail and give an overview of how it works. As we can see from Fig. 1, the architecture composes of two major models, Activity Recognition Model (ARM) and Signal Processing Model (SPM).

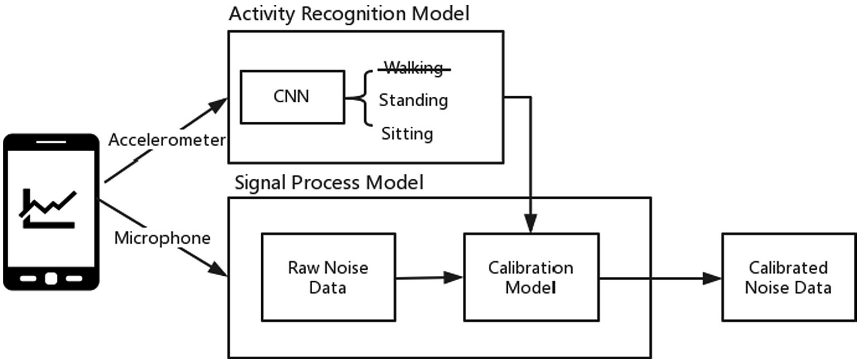


Fig. 1. The proposed architecture for noise sensing when mobile phones are in trouser pockets

There are two embedded sensors we will use, accelerometer and microphone. Firstly, the ARM detects the current activity of the user according to the accelerometer data based on CNN. When the user is in a correctable state (namely standing or sitting), the microphone is started to collect the acoustical signals, and the SPM module is started to calculate the decibel value of the surrounding environment, then the corresponding calibration sub-module is called to correct the initial measurements and output the calibrated values.

3.2 The Basis of Noise Level Measurement

We usually use the sound pressure level (SPL) to express the noise level and the A-frequency weighting sound level is the main standard for noise assessment today because it can reflect the loudness perceived by humans [14]. For evaluating the noise level in mobile phones, we must do a series of processing to compute these indicators. Since the frequency range that the human ear can sense is up to 20000 Hz, in our experiments, we collect audio signals at the rate of 44100 Hz for preserving all the audio signals information according to the Nyquist sampling theorem. We use the Android API of AudioRecord to collect the acoustic signals, whose parameter settings are displayed in Table 1.

Table 1. Android AudioRecord API parameters

Parameters	Value
Sampling rate	44100 Hz
Sample size	16 bit
Buffer size	4410 byte

The mobile phone collects samples at a frequency of 44.1 kHz, that is, 44100 samples are collected per second. Following the settings in Table 1, each sample has 16 bit thus one buffer can store 2205 samples and there will be 20 buffers produced with no overlapping per second.

In this paper, we use BENETECH GM1356 SLM to measure the actual noise level and compare with mobile phone measurements. GM1356 is compliant with standards of IEC PUB 651 TYPE2, whose time constants are 0.125 s for time-weighting F and 1 s for time-weighting S. We use its data storage function to record the noise level for a period of time. This function can measure the A-weighted, time-averaged sound level within 1 s with time-weighting S. But compared to the sampling rate of mobile phones, it's still much slower. Thus, we present a way to achieve data alignment, which is intuitively illustrated in Fig. 2.

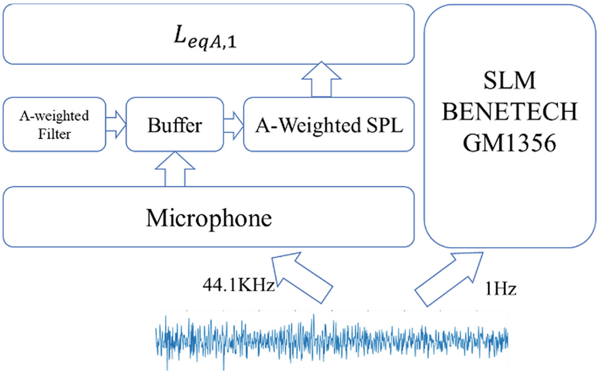


Fig. 2. Date alignment for solving different sampling frequency of SLM and mobile phone

After microphone collects audio signals at the rate of 44.1 kHz, the samples are stored in buffers temporally, we use an A-weighted filter [23] to obtain the A-weighted SPL in each buffer and calculate the A-weighted equivalent continuous sound level ($L_{eqA,T}$) [14] of the 20 buffers within 1 s. In this way, we can compare the measurements with the SLM readings by means of point to point of every second.

4 Experiments and Analysis

In this section, a series of experiments are carried out under different phone context. The mobile phones we used are MEIZU M5 Note, REDMI NOTE 5A and COOLPAD C106-9, and the standard SLM is BENETECH GM1356. Sensitivity is an important indicator of microphones, which is the ratio of the analog output voltage or digital output value to the input pressure [24] and the difference of microphones between mobile phones and SLM makes us have to do a series of processing and correction. In this paper, we define the hand-hold situation as the standard situation because it has been proven the feasibility and accuracy in many studies. All of our experiments are under the conditions of trouser pockets during sitting, standing or walking, except for the experiment in the standard situation.

4.1 Experiments for Constructing Calibration Sub-models

The experiments in this section are mainly for constructing calibration sub-models in our architecture, which will be integrated into SPM presented in Fig. 1.

Experiments in the Standard Situation. This experiment is carried out in the standard situation for proving mobile phones can accurately measure noise level in an acceptable range and comparing with the following experiments. In the experiment, we hold the phone and SLM closely and play the audio of traffic noise that was manually recorded before 1 m in front of them. This setting is for guaranteeing that both of them can simultaneously receive the acoustical signals.

The raw time series data collected from the mobile phone and SLM are described as Fig. 3.

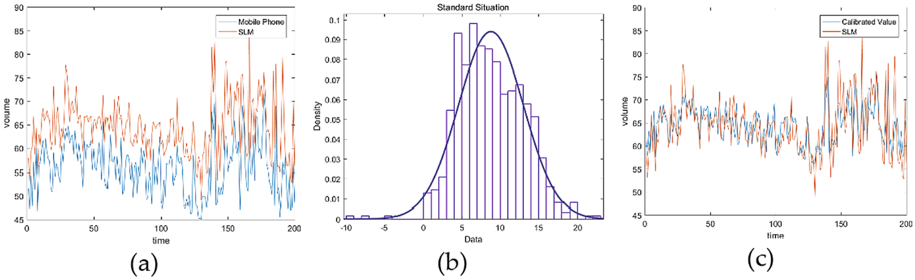


Fig. 3. (a) Time series data collected by the mobile phone; (b) Normal distribution fitting of the differences between measurements and SLM readings; (c) Time series data of standard SLM and mobile phone after linear model calibration (Color figure online)

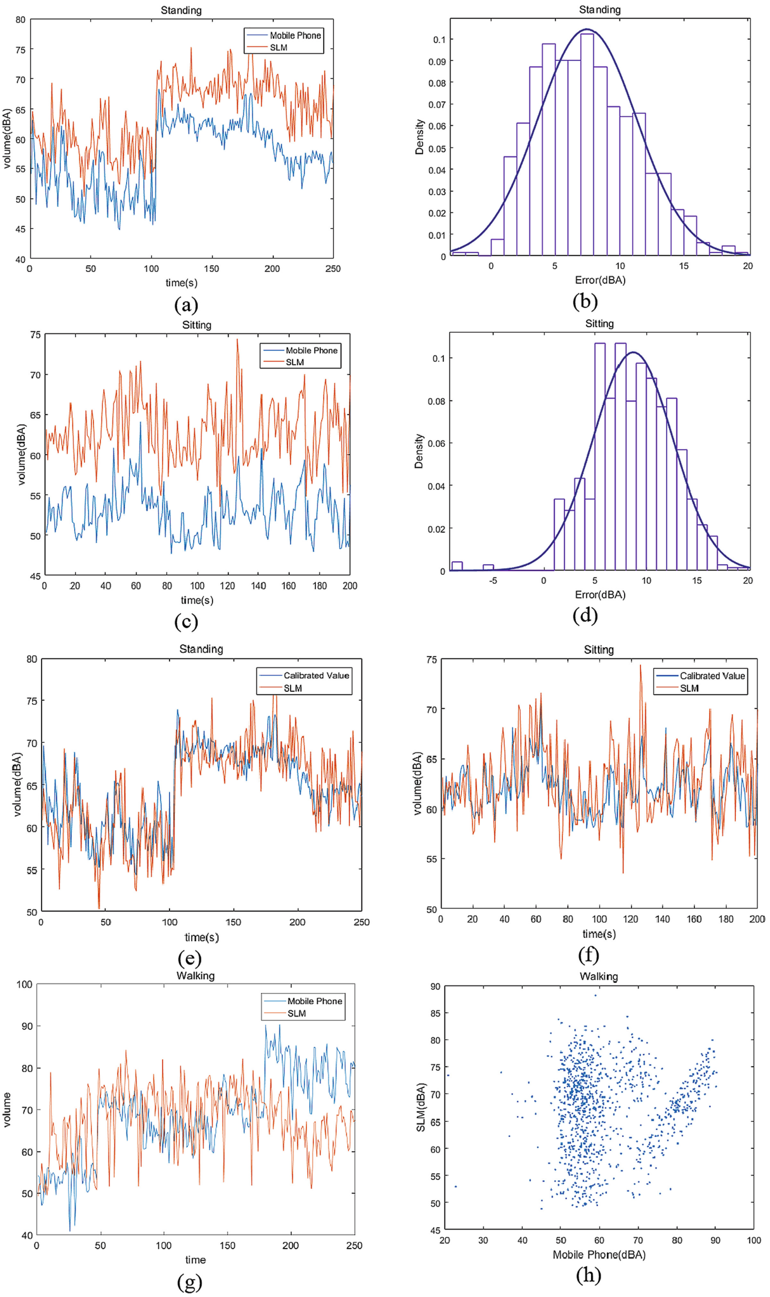
In Fig. 3(a), it's obvious that there exists a certain distance between SLM (orange line) and mobile phone (blue line), and SLM readings are always larger than mobile phone readings. For further analyzing the differences between them, we plot the histogram of original errors and investigate the data distribution. We can see that the errors are intensive in the range of 5–10 and basically conform to the normal distribution. Its normal distribution fitting is plotted with the blue line in Fig. 3(b) and the related parameters of the fitting result can be found in Table 2. Also, we try to do linear fitting between them and the coefficients of the obtained linear model is $y = 0.7534x + 22.29$. We adopt this model to calibrate the raw measurements, the calibration result can be intuitively seen from Fig. 3(c). The two lines generally overlap and the average error is about $\pm 2.32\,\mathrm{dB(A)}$. It's a satisfying result within an acceptable range because the 3 dB difference is imperceptible to the human ear [25], thus we decided to use the linear model to do the calibration.

Experiments in Other Situations. The ideal hand-hold phone context is not always available, in many situations, most people will choose to put phones in their trouser pockets. Considering that people are actionless and there is no friction between the phone and the pocket when they are standing or sitting, there may exist a slightly different calibrated model from the standard one because the microphone is in an enclosed space.

In this section, for verifying our assumption, we carry out a series of experiments under different phone context, sitting, standing, and walking when the phone is in the pocket. The experimental materials used in experiments are always kept the same, jeans pocket covered by the plaid shirt. The experimental results are intuitively depicted in Fig. 4.

The Fig. 4(a) and (c) are the raw time series signals collected by the mobile phone. We can directly see from them that there also exits a certain difference between them like the previous experiment and SLM readings are always larger than mobile phone readings. Thus, we calculate the differences between measurements and SLM readings whose histogram is plotted in the Fig. 4(b) and (d). The distributions of standing and sitting data fit with normal distribution better according to the log likelihood of −1806.52 and −2055.19, respectively. Furthermore, the lines of normal distribution fitting are depicted for better observing results and other indicators such as the mean and variance of fitting results are shown in Table 2. Compared with the results of the first experiment, obviously, the values of standing and sitting data are much closer to the standard situation than the walking data, which means the high possibility of correction. We analyze the differences not only in the time domain but also in the frequency domain under the standard situation, sitting and standing. Figure 5 shows the comparison lines of their 1/3 octave band spectrums under these three situations when receiving the same audio signals. These three figures are not exactly the same because they are in three different states. When sitting, due to the posture, the microphone of the mobile phone in the pocket is blocked, and the measurements are obviously lower than the other two cases. When standing, the difference is not so obvious, just because the mobile phone is in a closed space, slightly different from the standard situation. Although like this, the trends of sitting and standing are basically the same as the standard situation, which is consistent with the normal distribution fitting results.

Thus, we do the linear fitting to standing and sitting data similar to Sect. 4.1 and obtain respective coefficients. After linear fitting ($y = 0.8316x + 17.15$) of standing data, we get the result depicted in Fig. 4(e) whose average error is about ±2.90 dB(A). The raw data in Fig. 4(c) after the linear fitting ($y = 0.7911x + 20.02$) looks like Fig. 4(f) and the calibration model whose average error is within ±3.06 dB(A) is available. As for walking data, unfortunately, as we can see from its time series plot and scatter plot in Fig. 4(g–h), we can’t find a suitable relation and fitting model between walking data and SLM readings. The reason may be that the friction caused by users’ periodical motion when the phone is put in the pocket leads to deviant readings. And that’s why we decide to discard the walking data in our system architecture.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 4. (**a**) Raw time series of standing data; (**b**) Normal distribution fitting of raw errors of standing data; (**c**) Raw time series of sitting data; (**d**) Normal distribution fitting of raw errors of sitting data; (**e**) Time series of standing data after calibration; (**f**) Time series of sitting data after calibration; (**g**) Raw time series of walking data; (**h**) Scatter plot of SLM readings and raw data calculated by the mobile phone;

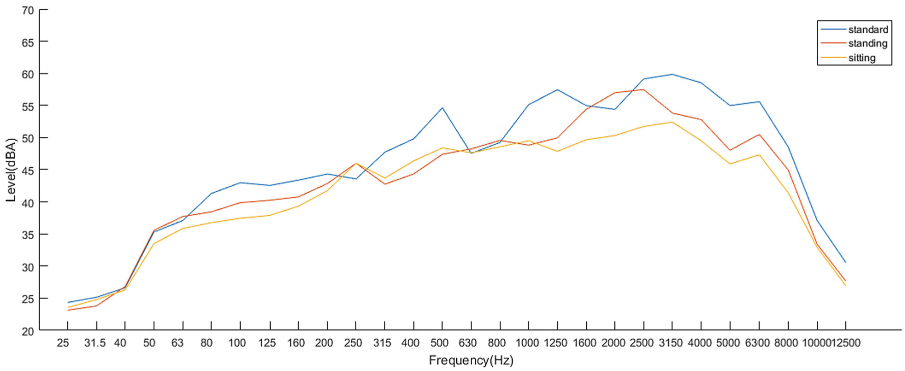


Fig. 5. The comparison lines under different situations (the lines are connected by the points corresponding to the values of 1/3 octave band spectrums under different situations, respectively)

Based on the above results, the standing calibration sub-model and sitting calibration sub-model are proved to be feasible and reasonable. So we integrate the two linear calibration sub-models into the SPM. After the integration, we can eventually conduct the experiments for the whole proposed architecture and validate its effectiveness in the next section.

Table 2. Results of normal distribution fitting

	Log likelihood	Mean	Variance
Standard situation	−1780.94	8.78134	17.9977
Standing	−1806.52	7.47372	14.5815
Sitting	−2055.19	8.71402	15.1505
Walking	−4146.17	5.20406	164.998

4.2 Experiments for Validating the Proposed Architecture

In this section, we carry out the experiment in the whole process based on the proposed architecture depicted in Fig. 1. In order to verify that there is no problem with device dependencies, we use three different types of mobile phones, MEIZU M5 Note, REDMI NOTE 5A, and COOLPAD C106-9 to validate the calibration model. Meanwhile, we use the classification models, support vector machine (SVM), k-nearest neighbor (KNN), and logistic regression (LR) to compare the performance with our CNN model.

Dataset. The dataset we collected from three mobile phones of different models includes 3-axis accelerometer data of x, y, and z sampled at 25 Hz and the raw original audio signals sensed by microphones at 44.1 kHz. The number of each type is shown in Table 3.

Table 3. The number of samples

	xyz-axis accelerometer	Audio signal
Sitting	24775	114357442
Walking	19371	197102180
Standing	17794	199931457
Total	61940	511391079

We collected 511391079 time series signal samples and 61940 3-axis accelerometer data in x, y, and z axis separately as the input data for training and testing the model. Firstly, we process the input 3-axis time series signals by our proposed CNN model and use it to classify and identify current user behaviors. Then, we leverage the calibration model obtained in the previous section to correct raw data and analyze the experimental results. More details will be explained in the next sub-sections.

The Proposed CNN Structure. Convolutional neural network (CNN) is one of the representative algorithms of deep learning. Since CNN can extract features from signals well and has the advantages of local dependency and scale invariance [26, 27]. We adopt the CNN model to process the raw time series accelerometer data following the idea of [26, 27] where the 3-axis data will be treated as the 3-channel input like RGB images. Given that we will integrate the classification model into mobile phones in the future, there is a tradeoff between computation and accuracy. In this paper, a relatively simple and classical CNN structure is adopted and the specific parameters are shown in Fig. 6.

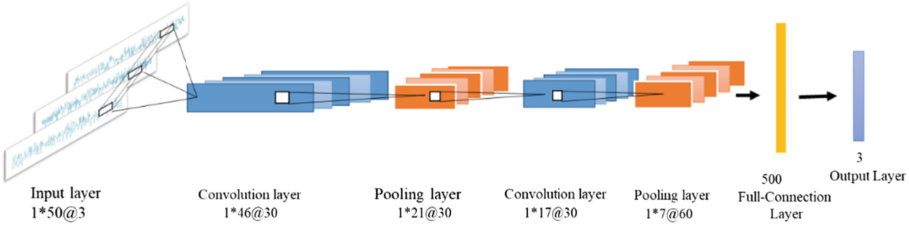


Fig. 6. The structure of CNN model used for classifying user activities, including the input layer, 6 hidden layers (between input layer and output layer), and output layer.

Due to the obvious numerical feature differences among these three activities walking, sitting and standing, the architecture we leveraged is quite simple but the classified result is very satisfying. The structure consists of 1 input layer, 2 convolution layers, 2 max-pooling layers, 1 full connection layer and 1 output layer. In our model, we directly take the time series data as input differing from traditional image input, so some transformations must be done before inputting. We set a sliding window with the length of 50 to segment the raw time series accelerometer readings. The window size is the sample length of 2 s, which is long enough to capture the repeated walking motion. The sliding window moves forward in a half-overlapping manner (namely 25 samples per step). Then segments are input into the input layer in 3-channel way. In the following convolution layer, a 1*5 filter (stride = 1) is adopted to conduct a one-dimensional convolutional operation. In pooling layer, we use max-pooling (size = 1*5, stride = 2) to reduce the dimensionality of feature maps in the previous layer. The operations and parameters in the next convolution layer and pooling layer are the same as these two's. The flattened last pooling layer is connected to a full-connection layer with 500 hidden units and the output layer will give the final classified result standing, sitting or walking.

Experimental Results. We use the 10-fold cross validation method to perform the classification model. The performance comparison of different classifier results is shown in Table 4. The precision, recall, and F1 of CNN classification results are the highest of all, which reach 99.2%, 99.1% and 99.1%, respectively. In conventional classifier of machine learning, KNN performs better than the other two and its precision is very close to CNN's.

Table 4. Performance comparison of different classifier results

	Precision	Recall	F1
CNN	0.99170	0.99140	0.99141
LR	0.98735	0.98710	0.98713
KNN	0.99083	0.99054	0.99057
SVM	0.98625	0.98538	0.98554

After classification, if it's walking, the data is abandoned directly. For the other two cases, we adopt corresponding calibration sub-models obtained in Sect. 4.1. The average errors of MEIZU, REDMI and COOLPAD finally obtained are ±3.0563 dB (A), ±3.0644 dB(A) and ±4.8252 dB(A), respectively. MEIZU and REDMI perform better than COOLPAD and control the errors in a similar range, about ±3 dB(A) which is an acceptable value. The results prove that the proposed calibration model can efficiently improve data quality under different phone context. Though the COOLPAD performs not as well as MEIZU and REDMI, it's already better than the original average error (±13.6740 dB(A)). More intuitively, let's take 120 samples before and after calibration as the example. The raw data of mobile phones and SLM are shown in Fig. 7(a). In the raw time series data, the first 60 samples are recorded during sitting while the following 60 samples are standing data.

In Fig. 7(a), all the lines of the three phones deviate from the SLM readings (blue line) at different levels and are smaller than the blue line, this is consistent with our previous experiments. As how the designed system architecture works, the three different lines are calibrated by sitting calibration sub-model and standing calibration sub-model, respectively. As we can see from the corrected result shown in Fig. 7(b), most points of MEIZU and REDMI are very close to the SLM readings and the orange and red lines generally overlap blue line. Although the purple line still has a small distance, it is much closer than Fig. 7(a). The differences among them after calibration mainly caused by the hardware configuration differences. The hardware configuration of

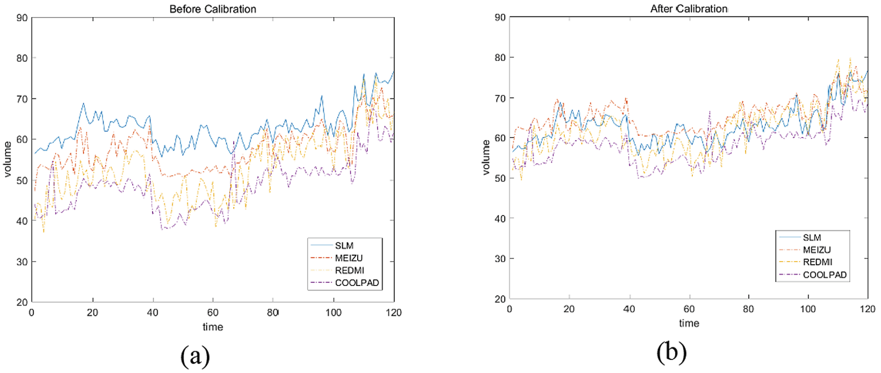


Fig. 7. Time series data, the first 60 samples are sitting data and the remaining is standing data (**a**) The raw series data before calibration; (**b**) The series data after calibration (Color figure online)

different brands and different models of mobile phones is quite different, and it will certainly affect the accuracy of the measured values to a certain extent. This is something we have no way to overcome at present but we will continue to study and solve it in the future work.

5 Conclusions

Noise pollution has obtained increasing attention in recent years with much more realization of its harm. But it's limited for us to know the specific sound strength, which is bad for monitoring ambient noise level. The previous work leveraging mobile sensors of mobile phones requires strict sampling conditions leading to little available data. The sparse data makes it difficult to reconstruct data for building up a noise map which can make people know the noise situation from a macroscopic perspective and provide much useful and meaningful information for the government.

In this paper, we conduct a sequence of experiments for validating feasibility in the standard situation and digging out the relations between phone readings and SLM readings during sitting and standing. The results show that linear fitting can reduce the differences between them and control the errors in the desirable range, ±2.09 dB(A) and ±3.06 dB(A) respectively. In Sect. 4.2, the calibration model and the CNN-based activity recognition classification are integrated into one to perform the whole noise sensing process. To avoid device dependent problem, we use three mobile phones of different models to conduct the experiments. In the first classification model, we leverage the 3-axis accelerometer data only but the accuracy reaches 99.2%. Then, output the classified result and choose a corresponding calibration sub-model to modify the raw noise data. Finally, we get the average error of ±3.06 dB(A), ±3.06 dB(A) and ±4.8252 dB(A), respectively. The results prove that the proposed noise sensing calibration model can fully leverage noise data under different phone context which is unavailable in previous work and relieve the data sparsity problem in some way.

Even though both REDMI and MEIZU perform well in the calibration model and obtain high-precision results, the COOLPAD works not as nicely as them. The main reason is the heterogeneous problem among different mobile phones that can't be ignored. The hardware configuration differences between different mobile phone models actually influence the model accuracy, and this is the major problem we must consider and solve in our future work.

References

1. China Environmental Noise Prevention and Control Annual Report (2017)
2. Zamora, W., Calafate, C.T., Cano, J.-C., Manzoni, P.: Noise-sensing using smartphones: determining the right time to sample. In: 15th International Conference on Advances in Mobile Computing and Multimedia, MoMM 2017, Salzburg, Austria, 4–6 December 2017, pp. 196–200. Association for Computing Machinery (2017)
3. Huang, M., Bai, Y., Chen, Y., Sun, B.: A distributed proactive service framework for crowd-sensing process. In: IEEE International Symposium on Autonomous Decentralized System, pp. 68–74 (2017)
4. Unsworth, K., Forte, A., Dilworth, R.: Urban informatics: the role of citizen participation in policy making. J. Urban Technol. **21**(4), 1–5 (2014)
5. Radicchi, A., Henckel, D., Memmel, M.: Citizens as smart, active sensors for a quiet and just city. The case of the "open source soundscapes" approach to identify, assess and plan "everyday quiet areas" in cities. Noise Mapping **4**(1), 1–20 (2017)
6. Picaut, J., et al.: Noise mapping based on participative measurements with a smartphone. Acoust. Soc. Am. J. **141**(5), 3808 (2017)
7. Aiello, L.M., Schifanella, R., Quercia, D., Aletta, F.: Chatty maps: constructing sound maps of urban areas from social media data. R. Soc. Open Sci. **3**(3) (2016)
8. Li, C., Liu, Y., Haklay, M.: Participatory soundscape sensing. Landsc. Urban Plan. **173**, 64–69 (2018)
9. D'Hondt, E., Stevens, M., Jacobs, A.: Participatory noise mapping works! An evaluation of participatory sensing as an alternative to standard techniques for environmental monitoring. Pervasive Mob. Comput. **9**(5), 681–694 (2013)
10. Cui, Y., Chipchase, J., Ichikawa, F.: A cross culture study on phone carrying and physical personalization. In: Aykin, N. (ed.) UI-HCII 2007. LNCS, vol. 4559, pp. 483–492. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73287-7_57
11. Al-Saloul, A.H.A., Li, J., Bei, Z., Zhu, Y.: NoiseCo: smartphone-based noise collection and correction. In: 4th International Conference on Computer Science and Network Technology, ICCSNT 2015, Harbin, China, 19–20 December 2015. Institute of Electrical and Electronics Engineers Inc. (2015)
12. Liu, L.: The design and implementation of a real-time fine-grained noise sensing system based on participatory sensing. Master, Shanghai Jiao Tong University (2015)
13. Liu, L., Zhu, Y.: Noise collection and presentation system based on crowd sensing. Comput. Eng. **41**(10), 160–164 (2015)
14. Zuo, J., Xia, H., Liu, S., Qiao, Y.: Mapping urban environmental noise using smartphones. Sensors **16**(10), 1692 (2016)
15. Kardous, C.A., Shaw, P.B.: Evaluation of smartphone sound measurement applications (apps) using external microphones - a follow-up study. J. Acoust. Soc. Am. **140**(4), EL327–EL333 (2016)

16. Rana, R., Chou, C.T., Bulusu, N., Kanhere, S., Hu, W.: Ear-Phone: a context-aware noise mapping using smart phones. Pervasive Mob. Comput. **7**(PA), 1–22 (2015)
17. Huo, Z.: Research and implementation of a crowdsensing-based noise map platform. Master, China University of Geosciences, Beijing (2016)
18. Kwapisz, J.R., Weiss, G.M., Moore, S.A.: Activity Recognition using Cell Phone Accelerometers. ACM SIGKDD Explor. Newsl. **12**, 74–82 (2011)
19. Wang, J., Chen, Y., Hao, S., Peng, X., Hu, L.: Deep learning for sensor-based activity recognition: a survey. Pattern Recogn. Lett. **119**, 3–11 (2017)
20. Ha, S., Yun, J.-M., Choi, S.: Multi-modal convolutional neural networks for activity recognition. In: IEEE International Conference on Systems, Man, and Cybernetics, SMC 2015, Kowloon Tong, Hong Kong, 9–12 October 2015, pp. 3017–3022. Institute of Electrical and Electronics Engineers Inc. (2015)
21. Rana, R., Chou, C.T., Bulusu, N., Kanhere, S., Hu, W.: Ear-Phone: a context-aware noise mapping using smart phones. Pervasive Mob. Comput. **17**, 1–22 (2015)
22. Miluzzo, E.., Papandrea, M., Lane, N.D., Lu, H., Campbell, A.T.: Pocket, bag, hand, etc. - automatically detecting phone context through discovery. In: First International Workshop on Sensing for App Phones at Sensys (2010)
23. Zamora, W., Calafate, C., Cano, J.C., Manzoni, P.: Accurate ambient noise assessment using smartphones. Sensors **17**(4), 917 (2017)
24. Lewis, J.: Understanding Microphone Sensitivity, 12 June 2018. https://www.analog.com/en/analog-dialogue/articles/understanding-microphone-sensitivity.html
25. Rana, R.K., Chou, C.T., Kanhere, S.S., Bulusu, N., Hu, W.: Ear-phone: an end-to-end participatory urban noise mapping system. In: 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN 2010, Stockholm, Sweden, 12–16 April 2010, pp. 105–116. Association for Computing Machinery (ACM) (2010)
26. Zeng, M., et al.: Convolutional Neural Networks for human activity recognition using mobile sensors. In: 2014 6th International Conference on Mobile Computing, Applications and Services, MobiCASE 2014, Austin, TX, USA, 6–7 November 2014, pp. 197–205. Institute of Electrical and Electronics Engineers Inc. (2015)
27. Yang, J.B., Nguyen, M.N., San, P.P., Li, X.L., Krishnaswamy, S.: Deep convolutional neural networks on multichannel time series for human activity recognition. In: 24th International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, 25–31 July 2015, pp. 3995–4001. International Joint Conferences on Artificial Intelligence (2015)