ACOUSTICS BASED PROXIMITY DETECTOR FOR MOBILE PHONES

Zacharie Cohen Andy Rodan Alon Eilam Pavel Lifshits
Signal and Image Processing Laboratory (SIPL)
Andrew and Erna Viterbi Faculty of Electrical Engineering, Technion - Israel Institute of Technology
Technion City, Haifa, Israel, https://sipl.eelabs.technion.ac.il

***Abstract*—Modern mobile phones are equipped with an infrared photoelectric proximity sensor, most commonly applied to turn off the touch screen during a phone call to prevent accidental touches when users' face/ear is detected in proximity to the screen. In this work we propose to achieve the sensing functionality without using a special sensor. Specifically, we use an already existing speaker and microphones for proximity sensing, without interfering with their originally intended operation. We build our method on the observation that the transfer function from the mobile phone speaker to the microphones varies as a function of objects located in the vicinity of the mobile phone.**

We have prototyped our solution on 2 mobile phones - Samsung Galaxy A70 and Xiaomi Mi 9, and achieved 99% accuracy of detection. We demonstrate that our method is robust to various environments, geometries, fabrics, and performs well in the presence of noise. We also evaluate the performance of various implementations, and discuss their trade-offs.

***Index Terms*—Acoustic Sensing, Mobile phone, Proximity sensor, Smartphone**

I. Introduction

A proximity sensor [1] is able to detect the presence of nearby objects without physical contact. It is usually implemented by emitting a beam of electromagnetic radiation (e.g., infrared), and sensing changes in the returned signal. The sensed object is referred to as a proximity sensor's target. Different target types require different sensors. For example, a capacitive proximity sensor or a photoelectric sensor are suitable for a plastic target, while an inductive proximity sensor only works with a metal target.

Nowadays, a proximity sensor is a standard component of most smartphones. Figure 1 shows an example of a proximity sensor in a Samsung smartphone located at the top near the front-facing camera. The main application of the sensor is to detect when a user is holding the phone near his/her face during a call, in order to turn off the display, to avoid an accidental touch during the call and to reduce the battery power consumption. Other applications may include saving power when the phone is in the pocket, or reduction of radiation exposure by attenuating radio power when the phone is in close proximity to the body.

In smartphones, proximity sensors are usually implemented by means of an IR LED and a light detector. However, such implementation has the following drawbacks: First, the need for a special sensor, which on top of the added cost and related circuitry, consumes both additional power and physical space on the screen, impeding a better screen-to-body ratio. Second,

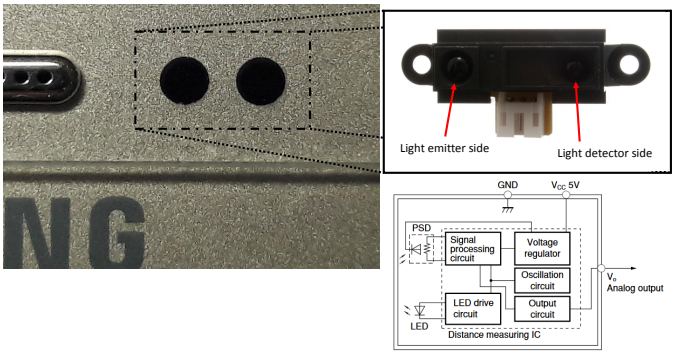

Fig. 1. An example of a proximity sensor in a Samsung smartphone and Sharp GP2Y0A21YK optical proximity sensor

improper installation of cases, covers or screen protectors may interfere with the proximity sensor function. Third, due to IR-based implementation, the performance of the sensor may vary over temperature and object color variations (e.g., failure to detect a close dark object because of the absorbing nature of the object color). Finally, such a sensor is inherently limited to detection in the area of the sensor, rather than proximity to the phone. For example, even partial occlusion by a finger would be detected as positive, while a large object even touching the screen below the sensor would not be detected (blind zone).

In this work we propose to utilize the **existing** speaker and microphones to implement an acoustic based proximity detector. We prototype our solution on 2 real mobile phones – Samsung Galaxy A70 and Xiaomi Mi 9. We show the robustness of our method to varying environments, geometries, fabrics, and to a variety of recorded noise signals. Moreover, we show evidence of a possibility to further extend the approach to achieve even finer grain sensing, such as, detecting the type of materials in proximity to the device and distance measurements.

Our method is based on the observation that the transfer function from the mobile phone speaker to the microphones varies as a function of objects located in the vicinity of the mobile phone [2]. We estimate the transfer functions and identify the transfer function's features which indicate an object proximity.

Such an approach, however, poses several challenges. First, we want to preserve the original functionality of the speaker

978-1-7281-8099-1/20/$31.00 ©2020 IEEE

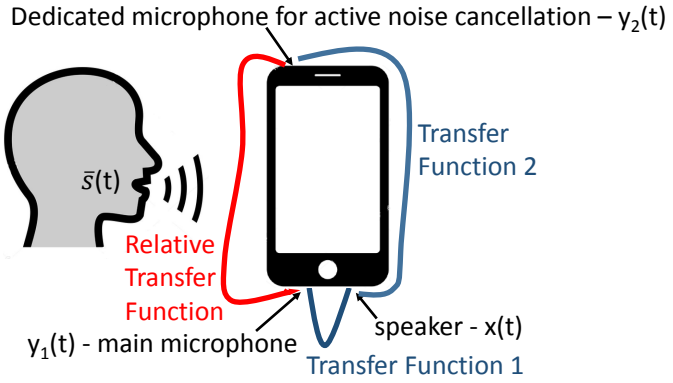

Fig. 2. Schematic diagram of a mobile phone with 2 microphones and definition of transfer functions

and microphones during the sensing period. Second, we have to operate with the given geometry, as well as the location of the speaker and the microphones. Finally, our detection needs to be robust to varying environments, materials, and noises. We address these challenges as follows. We use high frequency sound as means to estimate the transfer function, or speech signals as the means to estimate the relative transfer function (RTF) [3]–[5]. We consider a set of features and apply a feature selection algorithm to select only the ones that differentiate robustly between the states. We can use an SVM classifier [6] as the detector. As an alternative, we show how to train a convolutional neural network (CNN) directly on the transfer functions as inputs. The method is generic and modular.

We compare the performance of the different approaches to TF estimation and classifier implementations. We achieve accuracy of 99.3%, recall rate of 100%, and precision of 98.5%, with F1-score of 0.99, and an area under ROC curve of 0.99. Our additional contributions are a labeled dataset with 739 recordings, open-source Matlab/Python implementation of the algorithmic techniques used and an open-source Android application.

To the best of our knowledge, we are the first to suggest usage of an existing speaker and microphones of a mobile phone to perform proximity detection. Acoustic-based proximity and ranging is implemented by dedicated hardware (e.g., sonar). B. Thiel et al. [7] used an ultrasonic acoustic-based method of two mobile phones paired by bluetooth to estimate the distance between them. C. Peng et al. [8] uses an acoustic-based method for ranging and localization with dedicated hardware, while [9] uses beacons and unsynchronized ultrasonic sources to position a mobile phone. The authors of [2], [10]–[12] use acoustics to estimate the geometry of a room, using dedicated source signal, speaker, and microphone(s), however limiting the shape to a convex polyhedral room. [13] used harmonics produced in sound echoes to detect the material on which the smartphone was placed. Perhaps closest to our approach are US patents [14], [15], however, those either control the location of the loudspeaker and microphones, use dedicated speaker / microphone for the purpose of detection, or use audio in the human audible range.

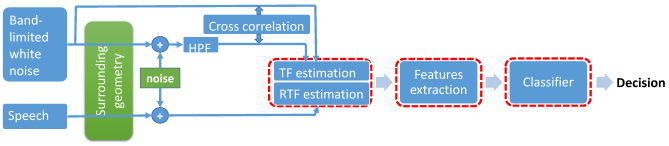

Fig. 3. Overview of proposed proximity detection method. The implementation of each block marked in red dashes may be done using different techniques

This paper is organized as follows. In Section II, we formulate the problem of proximity detection, and describe our proposed acoustic-based method. In Section III, we present experimental results to evaluate the performance and robustness of our method to environment, geometries, positioning and noise. Finally, in Section IV, we discuss future directions and conclusions.

II. SYSTEM OVERVIEW

The task of a proximity detection sensor is to detect whether an object is located close to the sensor. Our method suggests omitting the sensor, by utilizing the existing speaker and microphones, without compromising their functionality. A typical mobile phone is usually equipped with a speaker and two microphones, the locations of which are illustrated in Figure 2. One microphone is located at the bottom of the device and the other, which is commonly dedicated for active noise cancellation on top. We define three transfer functions: 1. TF between the speaker and the main microphone at the bottom of the phone 2. TF between the speaker and the microphone at the top of the phone, and 3. Relative transfer function (RTF) [3] between the microphones. We propose to estimate the TFs and to construct a classifier using a supervised machine learning approach. Our method consists of 3 subsequent stages, as depicted in Figure 3. Each stage can be implemented using different algorithmic techniques; the performance of each technique is evaluated in Section III and the trade-offs are discussed in Section IV. The first stage of transfer functions identification is described in §II-A. The second stage of feature extraction is described in §II-B. The final stage of classification is described in §II-C. A more direct approach combining the feature extraction and classification, using a convolutional neural network, is described in §II-D. In the training phase, the estimated TFs are labeled with an object in proximity or no objects in proximity, and are used to train SVM or CNN classifiers. Such trained classifier is used in the online classification phase to classify new unlabeled TFs as proximity detected or not-detected.

A. Transfer function identification

We propose two approaches for identification of the TFs. In the first approach, we estimate the TF using a synthetic input signal, in which case we estimate TFs 1 and 2. In the second approach, we estimate the RTF by using speech signals as a source.

Figure 3 illustrates the case of synthetic input signal $x(t)$; we use high frequency audio – a band-limited Gaussian white

noise in a frequency range of 15-20 KHz with a Gaussian envelope of length 1.5 sec, sampled at 44.1 KHz. Such a signal should not bother human users [2], or interfere with the speaker and microphone operation. Although the hearing threshold depends, among others, on age, the playback level with which the experiments have been performed, show that the signal lies well below the hearing threshold. We record the audio from both microphones at a sampling rate of 44.1 KHz, with a 16-bit resolution, apply a high-pass filter on the recorded signals with a cutoff frequency of 14.5 KHz to remove noise and irrelevant information. We then cross-correlate the recorded signals using the input signal to synchronize them. The transmitted sound interacts with the geometry of the surrounding objects; we model this interaction as a convolution system, with impulse response function $h_1(t)$, and with additive noise from the environment $n_1(t)$. The recorded signal that we analyze will then be:

$$y_1(t) = x(t) * h_1(t) + n_1(t)$$

Our goal is to extract $h_1(t)$ from the recorded sound. We use the least mean square (LMS) algorithm, introduced by [16], which is a popular method for adaptive system identification.

In the second approach, we define $s(t)$, as a non-stationary speech source signal, and $w_1(t)$ and $w_2(t)$ additive noise signals. We denote the signals received by the main microphone as

$$y_1(t) = s(t) + w_1(t)$$

and noise canceling microphones as

$$y_2(t) = h(t) * s(t) + w_2(t)$$

This time, our goal is to identify the response $h(t)$. Note that $s(t)$ is not a clean speech signal but a reverberated version, $s(t) = \overline{s}(t) * \hat{h_1}(t)$, where $\overline{s}(t)$ is the clean speech signal and $\hat{h_1}(t)$ is the surrounding impulse response of the main microphone to the speech source. Accordingly, $\hat{h_2}(t) = h(t) * \hat{h_1}(t)$ is the surrounding impulse response of the noise cancellation microphone to the speech source, and $h(t)$ represents the relative impulse response between the microphones with respect to the speech source. We closely follow the algorithm in [5] to identify the RTF.

B. Feature extraction and selection

Careful selection of features is crucial when designing an effective detector. For each transfer function, we seek features that comprise the essence of information relevant to classification, robust to conditions and noise, and are efficient in terms of computational complexity. To find such features, we have used a large set of features that were previously used in the literature for tasks of audio processing. The features we tried in the time domain are: Max, Min, Mean, Median, Standard Deviation, Root Mean Square, Averaged derivatives, Skewness, Kurtosis, Interquartile Range, Zero Crossing Rate, Mean Crossing Rate, Short time energy, max/mean ratio. In the frequency domain we tried: Mean, Standard deviation, Spectral flux, Spectral roll-off, Spectral centroid, and Spectral flatness.

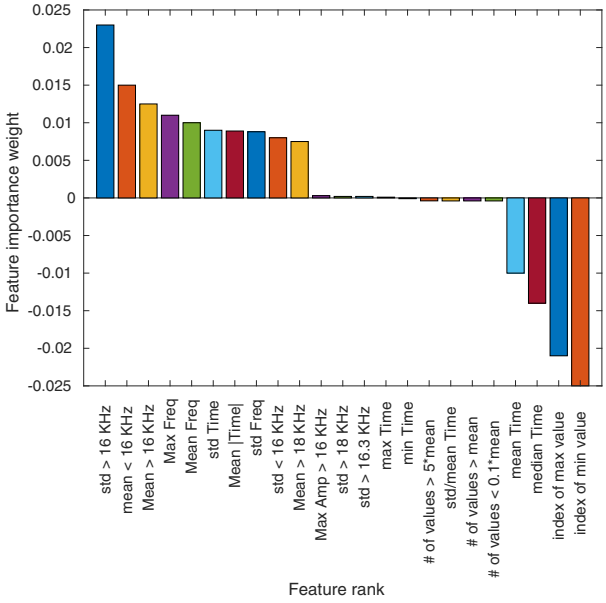

Fig. 4. Feature ranks and weights computed by ReliefF algorithm

Only some of the features described above may be useful for proximity detection. Feature selection is the process of selecting a subset of the features by removing redundant and irrelevant ones. This process reduces the dimensionality of the dataset, makes the models easier to interpret by researchers, enhances generalization by reducing variance and, in turn, reduces the computational and storage complexity of the classification and training time. We selected a subset of the features by using the ReliefF algorithm [17] to avoid exhaustive search. The feature ranks and weights computed by ReliefF on our training dataset are depicted in Figure 4. We can see in the figure that some features contribute to classification accuracy while others interfere. We find that using only the mean and standard deviation of the frequency response is enough to be able to linearly separate the cases; simplicity of such an approach at the expense of accuracy might make it worth considering.

C. SVM classifier

We trained a Support Vector Machine (SVM) [6] classifier. SVM classifies data by finding the hyperplane, with the largest margin (maximal width of the slab parallel to the hyperplane that has no interior data points) between the two classes that separates all data points of one class from those of the other class. We consider the standard linear as well as non-linear Gaussian (RBF) kernels. In all our experiments, the RBF kernel led to better results, typically by about 10%, compared to the linear kernel, when trained and tested using the same cross-validation process.

D. Convolutional Neural Network classifier

To avoid the manual craft of feature extraction and selection, we attempted to classify the transfer functions using a convo-

TABLE I
SUMMARY OF THE PERFORMANCE OF THE DETECTORS

		AUC	Recall	Precision	Accuracy	F1
LMS-based TF identification synthetic source	**SVM**	0.97	95.58%	98.18%	97.08%	0.97
	CNN	0.91	94.05%	94.05%	93.47%	0.94
RTF-based speech source	**SVM**	0.99	100%	98.51%	99.29%	0.99
	CNN	0.81	83.33%	81.15%	82.29%	0.82

lutional neural network (CNN). We constructed a CNN with a typical topology [18]: three convolutional layers followed by two fully connected layers and a softmax layer with two outputs. Each convolutional layer has 64 1x5 filters followed by 50% pooling, ReLU activation, batch normalization, and 20% dropout. The network is trained end-to-end using the cross-entropy loss function. We limit our model size to having the number of parameters in the order of magnitude of the number of training sample points to avoid over-fitting. In the case of RTF-based approach, due to limited availability of training data, we initiated the weights of the model with the previously learned model for TF, as often done in transfer learning.

III. EVALUATION

We have examined our approach in several setups with two mobile phones of different make and model. We have collected recordings of generated high freq white noise played by the phone, having objects such as human face or body, table, book, ceramic sink, transparent glass window, sofa and wall in close proximity (less than 10 cm), and without any objects, in real environments. Recordings of a person speaking during the conditions described above were collected as well. For that purpose we introduce in this work a new dataset of recordings. Our dataset contains 739 audio records (353 positive, having objects in proximity, and 386 negative - without objects in proximity); 480 were recorded by Samsung Galaxy A70 and 259 were recorded by Xiaomi Mi 9. Of the positive recordings, 280 are of high frequency white noise and 73 are of speech. Of the negative recordings, 320 are of high frequency white noise and 66 are of speech. Recordings with objects in proximity have additional labels as distance, and type of material in proximity. We'll make our dataset available for download at the labs' website.

To evaluate the performance of our techniques, unless stated otherwise, we used 5-fold cross-validation. We use 60% of the dataset to train the model, 20% for validation and parameter tuning and 20% for testing. To evaluate the effect of noise, we used eight noise types – suburban train noise, crowd of people (babble), car noise, exhibition hall, restaurant, street, airport and train-station noise [19], [20], with SNR values ranging from 0 dB to 20 dB.

Figure 5 shows the receiver operating characteristic curve for the proposed techniques. Table I summarizes the results of the performance of the detectors, comparing different algorithmic approaches. We can see that the RTF-based method is

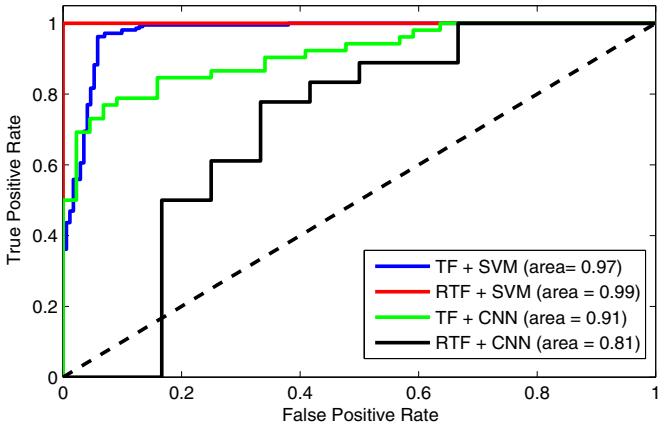

Fig. 5. Receiver operating characteristic curve for implementation methods

superior in performance to the LMS-based channel estimation. However, as we show in Figure 6, the performance of the RTF-based method is affected by noise; thus in the presence of noise, using high frequency white noise to estimate the TF, is preferred. While CNN-based method slightly underperformed the SVM-based method, it still has the following advantages: 1. Not having to manually craft features; 2. An expectation of substantial performance improvement as a function of availability of labeled data; 3. The existence of dedicated hardware accelerators for CNN-based inference. All the above indicates that the CNN-based method should still be considered as a viable option.

To evaluate the robustness of the classifier to varying environments, geometries, and positioning, we constructed a test set (a subset of the dataset) which contains samples with previously unseen (in the training phase) material/environment/geometries. In all the experiments concluded, the results were within 2% of the results present in Table I. This demonstrates the robustness of our method.

Figure 6 shows the AUC as a function of the SNR of the above-described noise types for the SVM-based classification with an estimated TF or RTF. While the performance of RTF-based method deteriorates in the presence of noise, the method using a synthetic source shows resilience (even at the low SNR of 5 dB), due to the fact that the evaluated noise types contain little or no energy above 15 KHz.

To examine the real-life performance of our proposed method, we developed an Android application which plays the synthetic high frequency white-noise, records the output from both microphones and uses MATLAB C code generator in combination with Android NDK to execute the pre-trained model directly on the phone. During a crowded project-demo day at our faculty, 27 tests were conducted and resulted in 100% successful classifications. We'll make our application available for download at the labs' website.

IV. CONCLUSIONS AND FUTURE WORK

In this work, we have shown a method for an acoustic-based detection of objects in proximity to the mobile phone, using

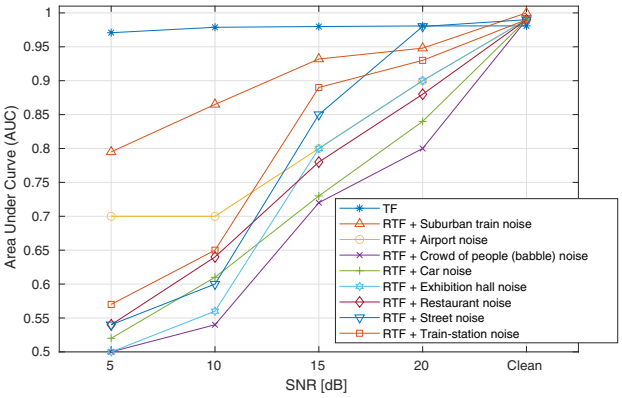

Fig. 6. Performance of detectors as a function of noise

existing hardware while allowing its original functionality during the detection, thus obviating the need for a special sensor. Such an implementation overcomes the drawbacks of existing IR-based technology, most notably the blind zone, and false-positive detection due to partial occlusion. It allows for energy savings, as during a phone call the microphones and speaker are already being used, so the additional energy consumption comes only from the required computation, which can be performed on a low power DSP, usually placed on the audio path for tasks like noise cancellation and speech coding. Cost savings and the extension of screen-to-body ratio, can be achieved as well, by exclusion of the dedicated sensor. Moreover, it is possible to achieve finer grain sensing, such as ranging, material detection, or controlling the area of detection, by training additional specific models for the condition of interest. Our preliminary experimental results, showing clustering by material type in low-dimensional t-SNE embedding [21] of the TFs plot, suggest an interesting future direction into extension of the classifiers for finer grain acoustic-based sensing of type of materials. Another future direction is further improvement of the algorithms to allow sensing in shorter time frames.

V. Acknowledgement

The authors are grateful to DSP Group, Inc. and in particular to Lior Blanka and Dima Lvov for their kind support of this work and for exploring various use-cases of its implementation. The authors would also like to thank Prof. Mark Silberstein, head of Accelerated Computer Systems Lab (ACSL), Prof. David Malah, head of Signal and Image Processing Laboratory (SIPL), Prof. Ronen Talmon, Nimrod Peleg, SIPL's chief engineer, and Yair Moshe for their support, advice, and helpful comments.

References

[1] "IEC 60947-5-2: Low-voltage switchgear and controlgear - Part 5-2: Control circuit devices and switching elements - Proximity switches," 2012.

[2] I. Dokmanić, R. Parhizkar, A. Walther, Y. M. Lu, and M. Vetterli, "Acoustic echoes reveal room shape," *Proceedings of the National Academy of Sciences*, vol. 110, no. 30, pp. 12 186–12 191, 2013.

[3] S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Transactions on Signal Processing*, vol. 49, no. 8, pp. 1614–1626, 2001.

[4] I. Cohen, "Relative transfer function identification using speech signals," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 451–459, 2004.

[5] R. Talmon, I. Cohen, and S. Gannot, "Relative transfer function identification using convolutive transfer function approximation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 546–555, 2009.

[6] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[7] B. Thiel, K. Kloch, and P. Lukowicz, "Sound-based proximity detection with mobile phones," in *Proceedings of the Third International Workshop on Sensing Applications on Mobile Phones*, ser. PhoneSense '12. New York, NY, USA: ACM, 2012, pp. 4:1–4:4. [Online]. Available: http://doi.acm.org/10.1145/2389148.2389152

[8] C. Peng, G. Shen, and Y. Zhang, "BeepBeep: A High-accuracy Acoustic-based System for Ranging and Localization Using COTS Devices," *ACM Trans. Embed. Comput. Syst.*, vol. 11, no. 1, pp. 4:1–4:29, Apr. 2012. [Online]. Available: http://doi.acm.org/10.1145/2146417.2146421

[9] G. Feferman, M. Blatt, and A. Eilam, "Indoor positioning with unsynchronized sound sources," in *2018 IEEE International Conference on the Science of Electrical Engineering in Israel (ICSEE)*, Dec 2018, pp. 1–4.

[10] M. Crocco, A. Trucco, V. Murino, and A. Del Bue, "Towards fully uncalibrated room reconstruction with sound," in *2014 22nd European Signal Processing Conference (EUSIPCO)*. IEEE, 2014, pp. 910–914.

[11] I. Dokmanic and M. Vetterli, "Listening to distances and hearing shapes: Inverse problems in room acoustics and beyond," *EPFL, Lausanne*, 2015.

[12] F. Peng, T. Wang, and B. Chen, "Room shape reconstruction with a single mobile acoustic sensor," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2015, pp. 1116–1120.

[13] T. Hasegawa, S. Hirahashi, and M. Koshino, "Determining a smartphone's placement by material detection using harmonics produced in sound echoes," in *Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2016, pp. 246–253.

[14] V. Myllyla, "Acoustical proximity detection for mobile terminals and other devices," Apr. 1 2003, US Patent 6,542,436.

[15] S. Mattisson, "Sound-based proximity detector," Aug. 28 2007, US Patent 7,263,373.

[16] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Prentice-hall Englewood Cliffs, NJ, 1985, vol. 15.

[17] I. Kononenko, E. Šimec, and M. Robnik-Šikonja, "Overcoming the myopia of inductive learning algorithms with relieff," *Applied Intelligence*, vol. 7, no. 1, pp. 39–55, 1997.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[19] H.-G. Hirsch and D. Pearce, "The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*, 2000.

[20] H. G. Hirsch, "Fant-filtering and noise adding tool," *Niederrhein University of Applied Sciences, http://dnt.-kr. hsnr. de/download. html*, 2005.

[21] L. v. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.