LOW POWER ULTRASONIC GESTURE RECOGNITION FOR MOBILE HANDSETS

Costas Yiallourides, Pablo Peso Parada**

*Cirrus Logic Inc, London, UK. Email: pablo.pesoparada@cirrus.com

ABSTRACT

A novel approach for gesture recognition on mobile handsets which does not require any additional transducers is presented. The method is based on transmitting ultrasonic pulses from the earpiece and the loudspeaker and receiving with two microphones, usually located at the top and the bottom of the handset. Signal to Noise Ratio estimates are computed from the reflected signals on each microphone from which statistical moments are extracted and used for training a Support Vector Machine classifier along with hyperparameter optimization. The accuracy achieved is 77.5% on a database of 400 observations using 5-fold cross-validation.

Index Terms— gesture recognition, classification, support vector machines, ultrasound, handsets

1. INTRODUCTION

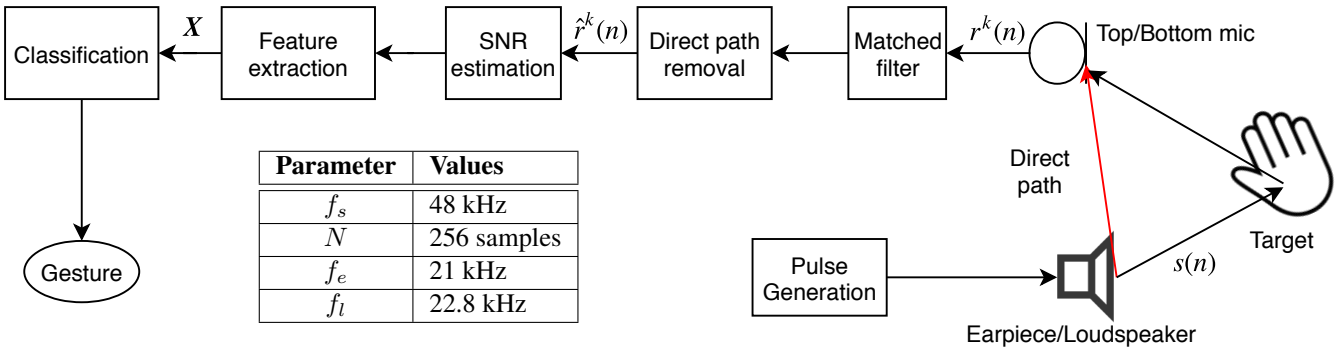
The mobile market is ever increasing. The penetration rate of unique mobile subscribers in 2017 was 66% of the total population, 57% of which were smartphone users, and is predicted to reach 71% in 2025 (5.9 billion users) according to [1]. The huge potential of this market led researchers and companies to develop sophisticated speech recognition solutions to offer an alternative method for user-smartphone interaction. However, little attention has been given into creating product solutions of lower complexity that can enhance the user experience.

Gestures are a natural way of using smart mobile devices. Commercially available solutions rely on computer vision [2–4]. These techniques are prone to low light intensity near the device and to the camera's view range. This results in poor quality of interaction which inevitably disheartens the users from using their devices as seamlessly as they would like to.

Motivated by this, researchers focused on ultrasound based gesture recognition that avoids the limitation of camera-based methods. Algorithms based on the Doppler Effect (DE) have emerged as the standard approach [5–7]. DE is a well-known phenomenon which characterises the frequency change of a wave for an observer who is moving relative to the wave source [8]. In [5] the researchers used the laptop's speakers and microphones to continuously transmit a tone signal of 18-22 kHz and track the DE caused by hand movements. The target's speed, direction, proximity, size and the variation of these over time were used as features to detect gestures like single tap and scrolling by setting threshold values. The DE is also utilized in [6] where a single microphone and the loudspeaker of a smartphone which transmits a continuous 21 kHz tone, are used. The sequence of frequency shifts obtained from the time frames, after performing a 4096-point Fast Fourier Transform, are used for classification. In [9] the authors aimed at detecting gestures performed inside a triangular space formed by three receivers and a transmitter emitting a 40 kHz tone signal and placed in the centre. By using 60 features derived from principal component analysis applied on cepstral coefficients and a Bayesian classifier, they report a recognition accuracy of 88.42% on a set of eight gestures. In [7], tone signals (>20 kHz) are continuously transmitted from external speakers attached on a mobile device. Duration and speed related features are extracted based on the DE and distance based features from the estimated impulse response. An average recognition accuracy of 92.6% for eight gestures obtained with a decision tree was reported. A sensor was developed in [10], specifically for gesture detection and recognition. However, it uses miniature radar technology and is therefore not compared with our work. Other researchers focused on multi-device interaction techniques [11, 12] which is outside the scope of this work.

Many of these approaches require customized hardware [9, 13, 14] or have high computational complexity and cost [5, 15]. These reasons make an algorithm impractical and prohibitive for real-time use on mobile handsets as it can drain the battery life quickly. Therefore, there is a need for low power, robust and accurate gesture recognition technique.

The work presented in this paper focuses on the formulation of a gesture recognition algorithm that would satisfy these requirements. It has an extremely low complexity as all the processing is done in the time domain. It is also robust due to averaging on the received block to compute the Signal to Noise Ratio (SNR) and to the use of statistics instead of making a decision on a per signal block basis. The algorithm uses only the built-in handset's speakers and microphones and the aim is to detect the changes in the echo signal that will determine the gesture made. The interest in this work is to identify the following gestures: (a) top to bottom (TB), (b) bottom to top (BT), (c) right to left (RL) and (d) left to right (LR) with the hand passing over the phone. These were chosen as they are intuitive to the user and can be mapped to anything from scrolling and swiping to answering and declining calls.



Parameter	Values
f_s	48 kHz
N	256 samples
f_e	21 kHz
f_l	22.8 kHz

Fig. 1. Block diagram of the proposed algorithm with the table of parameter values.

2. ALGORITHM

The proposed method operates at a sampling frequency f_s of 48 kHz and is based on extracting a set of features from the reflections of the transmitted ultrasonic pulses. These pulses are transmitted through the earpiece and the loudspeaker. If a target is present, the echo signal is captured by two microphones usually located at the top and the bottom of the handset. Fig. 1 shows the block diagram of the proposed algorithm where only one transmitter and receiver are shown for simplicity. Let

$$s_e(n) = \sin(2\pi f_e n)w(n) \quad (1)$$

$$s_l(n) = \sin(2\pi f_l n)w(n) \quad (2)$$

denote the transmitted pulses from the earpiece and loudspeaker respectively, where f_e and f_l are the pulse frequencies with $f_e \neq f_l$, N is the pulse length, $n = 1, 2, \ldots, N$ and $w(n)$ is a symmetric Nuttall defined minimum 4-term Blackman-Harris window as in [16]. These signals are transmitted every $8N$ samples in order to allow enough time to receive and process the echo before the next one and to reduce power consumption. On the receiver end, the k^{th} received signal block is denoted as $r_t^k(n)$ and $r_b^k(n)$ where the subscripts t and b refer to top and bottom microphone respectively. These are then processed as

$$\hat{r}_t^k(n) = \sum_{m=-M/2}^{M/2-1} r_t^k(m)s_e(n-m) - \hat{r}_t^{k-1}(n) \quad (3)$$

$$\hat{r}_b^k(n) = \sum_{m=-M/2}^{M/2-1} r_b^k(m)s_l(n-m) - \hat{r}_b^{k-1}(n) \quad (4)$$

where $M = 3N$ in which the delay due to the convolution operation is taken into account. The subtraction of the previously processed block is done in order to remove the direct path signal, transmitted through the phone frame, which is uninformative with respect to the gesture and is assumed to be constant between successive blocks.

The performance for frequencies greater than 20 kHz is not guaranteed given that the smartphone's microphones and speakers are designed for speech and audio and that the higher f_e and f_l are, the greater the attenuation due to the propagation through air is [17]. Therefore, each microphone signal is filtered with the time reversed coefficients (matched filter) of the pulse transmitted from the closest speaker, aiming at filtering out the spectral content that is outside the frequencies of interest.

2.1. Feature extraction

The processed signal blocks $\hat{r}_t^k(n)$ and $\hat{r}_b^k(n)$ are then used to obtain two SNR estimates, one for each microphone, as the ratio of $\frac{1}{e_2-e_1+1}\sum_{m=e_1}^{e_2}(e(m))^2$ to $\frac{1}{\nu_2-\nu_1+1}\sum_{m=\nu_1}^{\nu_2}(\nu(m))^2$ where $e(m)$ and $\nu(m)$ are the echo (green) and noise (red) regions respectively in Fig. 2 and e_1, e_2, ν_1, ν_2 are the indices denoting the start and end samples of the regions. $e(m)$ is the region where user activity is expected and is constrained to the first 24% of $\hat{r}^k$, after compensating for the filter delay. This gives a detection range of 44 cm (sound speed in air is 343 m/s) whereas $\nu(m)$ is constrained in the last 25%.

Considering the series of such SNR estimates with Z samples in total per microphone, we can think of it as a probability density function $\boldsymbol{P}$ as shown in Fig. 3. Its shape is of interest and by defining $\boldsymbol{g}_m^{\mu_0} = \left[(1/Z-\mu_0)^m, (2/Z-\mu_0)^m, \ldots, (Z/Z-\mu_0)^m\right]$ as the vector giving the position of an element in the series, the first four statistical moments are computed as

$$\mu = \boldsymbol{g}_1^0\boldsymbol{P}^T, \sigma^2 = \boldsymbol{g}_2^{\mu}\boldsymbol{P}^T, s = \frac{\boldsymbol{g}_3^{\mu}\boldsymbol{P}^T}{\sigma^3}, \kappa = \frac{\boldsymbol{g}_4^{\mu}\boldsymbol{P}^T}{\sigma^4} \quad (5)$$

where $\boldsymbol{P}$ is normalized such that $\sum_{z=1}^{Z-1} P_z = 1$. By defining $\boldsymbol{g}_m^{\mu_0}$ in this way, it ensures that the duration of the gesture does not affect the values of (5) as its maximum value is limited to 1. Moreover, the set of moments captures the temporal information which can be important for classification as can be seen in Fig. 3 where the SNR estimates from the bottom mi-

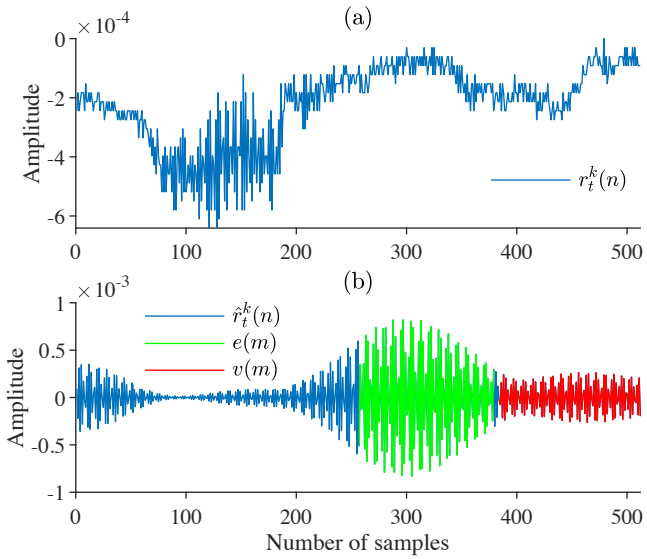


Fig. 2. Example of (a) $r_t^k(n)$ and (b) $\hat{r}_t^k(n)$ with the echo and noise regions where $N = 256$ and block size is 512.

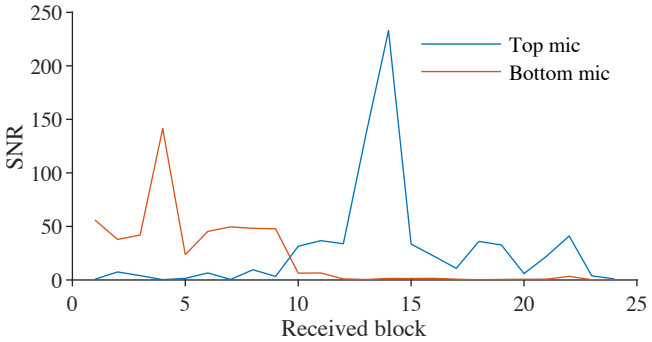


Fig. 3. SNR estimates for a bottom to top gesture.

crophone have higher values at the first few received blocks where the hand is closer to that microphone. For a single gesture, the feature vector is obtained as an 8-dimensional vector $\boldsymbol{x}^i = [\mu_t, \sigma_t^2, s_t, \kappa_t, \mu_b, \sigma_b^2, s_b, \kappa_b]$ with the subscripts denoting the microphone location and $i = 1, 2, \ldots, c$ the gesture class number. The feature set is then constructed as $\boldsymbol{X} = [\boldsymbol{x}_1^i, \boldsymbol{x}_2^i, \ldots, \boldsymbol{x}_L^i]^T$ for L recordings.

2.2. Classification

Assuming the use of a gesture detection algorithm, $\boldsymbol{x}^i$ can be used to identify the gesture class. This is obtained by the decision boundaries formed by the Support Vector Machine (SVM) approach [18]. For classes i, j with Q and $\mathbb{Q}$ number of observations respectively, the SVM in its dual form is given by [19]

$$\min_{\alpha} \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{B}\boldsymbol{\alpha} - \boldsymbol{e}^T\boldsymbol{\alpha} \text{ subject to: } \boldsymbol{y}^T\boldsymbol{\alpha} = 0 \qquad (6)$$

with $0 \leq \alpha_p \leq C\ \forall p$, $B_{qq} = y_q^i y_q^j K(\boldsymbol{x}_q^i, \boldsymbol{x}_q^j)$, $K(\cdot,\cdot)$ is a kernel, $\boldsymbol{e} = [1, 1, \ldots, 1]^T$, $q = 1, 2, \ldots, Q$, $\boldsymbol{q} = 1, 2, \ldots, \mathbb{Q}$, $p = 1, 2, \ldots, P$ with $P = Q + \mathbb{Q}$, $\boldsymbol{x}_q^i$ and $\boldsymbol{x}_q^j$ are the training vectors obtained from $\boldsymbol{X}$ with corresponding labels y_q^i and y_q^j. In this work, a one vs one approach was followed in which $c(c-1)/2$ classifiers are constructed where each one is trained with data from two classes [20]. For classification, a majority vote strategy was used based on $\text{sign}(\sum_{p=1}^{P} y_p^{i,j}\alpha_p K(\boldsymbol{x}, \boldsymbol{x}_p^{i,j}) + b^{i,j})$ where b is the hyperplane intercept point. In this formulation only the $\boldsymbol{x}_p^{i,j}$ of $\alpha_p > 0$ are used which are the support vectors essentially. For $K(\cdot,\cdot)$, four different functions are considered [21]:

- Linear (K_1): $K(\boldsymbol{x}_q, \boldsymbol{x}_q) = \langle \boldsymbol{x}_q, \boldsymbol{x}_q \rangle$ (7)
- Polynomial (K_2): $K(\boldsymbol{x}_q, \boldsymbol{x}_q) = (\gamma\langle \boldsymbol{x}_q, \boldsymbol{x}_q \rangle)^d$ (8)
- Sigmoid (K_3): $K(\boldsymbol{x}_q, \boldsymbol{x}_q) = \tanh(\gamma\langle \boldsymbol{x}_q, \boldsymbol{x}_q \rangle)$ (9)
- Gaussian Radial basis function (RBF) (K_4):

$$K(\boldsymbol{x}_q, \boldsymbol{x}_q) = \exp(-\gamma \left\| \langle \boldsymbol{x}_q, \boldsymbol{x}_q \rangle \right\|^2) \qquad (10)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of the two vectors. To this date, there is no scientifically proven optimization technique for finding the best kernel as it is depends on the data and the application. In this paper, a practical approach was followed in choosing the most suitable kernel along with the best values for the set of hyperparameters $\Lambda = \{C, \gamma, d\}$. This is described in Section 3.2.

We are interested in recognizing 4 gestures and a detection algorithm is therefore assumed to be used. However, this algorithm might not be perfect and in order to increase the robustness of the classifier in the proposed approach and make it less prone to false positives, a fifth class is also considered and is defined as the background noise recordings in the database.

3. EVALUATION

3.1. Experimental setup

For SVM model training and the algorithm evaluation, 360 gestures performed by 9 users, and 40 noise observations were used. Sound data was captured using a Samsung Galaxy S6 handset and the following procedure: The user places the handset flat on the table and then performs an instance of one of the 4 gestures stated in Section 1 with one hand while with the other starts and stops the data recording on the laptop. In this way the transition between the active state (i.e. gesture) and idle state, and vice versa, is avoided, which can be subjective and could lead to ambiguous decisions. More recordings are then carried out in the same manner for each of the four types and a total of 90 observations (22.5% of the database) for each gesture type are obtained. The gestures were recorded in a quiet room and the background noise of the same room as well as that of a noisy office were also obtained. The algorithm parameter values used are indicated in the table of Fig. 1.

3.2. Methods

Given that the task is multi-class classification, the algorithm's performance is evaluated based on the average of the F_1 scores of each class weighted by the number of true instances per class and is denoted as $\overline{F}_1$. Cross validation testing using 5 folds in conjunction with grid search for hyperparameter optimization were conducted. During cross validation the features of the training folds were standardized by subtracting the mean and scaling to unit variance. The same scale values are applied to the test fold. Table 1 shows the hyperparameters and the range of values tested. For the linear kernel the only relevant parameter is C while d is only used with the polynomial kernel.

Parameter	Values
$K(\cdot,\cdot)$	K_1, K_2, K_3, K_4
C	[1, 2, ..., 20, 30, 40, ..., 100, 1000]
γ	[5, 2, 0.125, 10^n] for $n = 1, 0, \ldots, -6$
d	[2, 3, 4, 5]

Table 1. SVM hyperparameters to be optimized.

3.3. Results

Table 2 summarises the best results obtained per kernel. The models are ranked based on the average cross validation $\overline{F}_1$ score and the rank column gives their overall rank compared to all the trained models. The SVM with RBF kernel, $C = 70$ and $\gamma = 0.01$, achieved the highest average $\overline{F}_1$ equal to 0.771 (with an equivalent average accuracy of 77.5%). Fig. 4 shows the confusion matrix obtained with this model. The most common error is the LR class which gives a class accuracy of 48.9%. On the contrary, for the BT and noise classes the accuracy is 90% while for the remaining two it is at least 80%.

The number of support vectors is related to the generalization performance of the classifier. The smaller the number the better the generalization is expected to be [22]. This could explain the lower accuracy achieved for the LR class as the support vectors found per class are TB: 19, BT: 29, RL: 28 and LR: 73. However, there also might exist some overlap in the feature space between the LR and RL classes. The results nevertheless, suggest that the proposed features carry significant discriminatory information and the trained model will generalize with high accuracy on unseen data.

The proposed method is compared against the one described in [5] which is one of the most cited ultrasound based gesture recognition methods. In this approach, fixed thresholds for detecting and recognizing gestures are used but here we find more general thresholds given the features by using SVM. The setup and methods described in Sections 3.1 and 3.2 are used and a linear SVM with C parameter optimization using the features from the baseline method is trained. The features in [5] are per signal block k and therefore, to make a fair comparison to the results of the proposed method we assume that each block has the same label as the gesture it belongs to and then train and test the classifier per block. A per gesture accuracy is obtained by performing majority voting on each gesture's predicted block class. The results obtained indicate poor discriminatory power of these per signal block features for 3 out of the 5 classes as most of the gestures are classified as noise. In summary, comparing with the best results of Table 2, the baseline accuracy is 31.5% vs 77.5% and $\overline{F}_1$ is 0.206 vs 0.771. The performance of the baseline is affected by the low amplitude of the transmitted pulses which further supports the robustness of our proposed set of features.

$K(\cdot,\cdot)$	C	γ	d	mean $\overline{F}_1$	std $\overline{F}_1$	Accuracy	rank
K_1	17	-	-	0.744	0.070	75.3%	50
K_2	40	0.10	3	0.735	0.049	73.8%	89
K_3	70	0.01	-	0.747	0.069	75.5%	45
K_4	70	0.01	-	0.771	0.084	77.5%	1

Table 2. Best average cross-validation training results for each kernel. The last column gives the rank of each model compared to all the trained models.

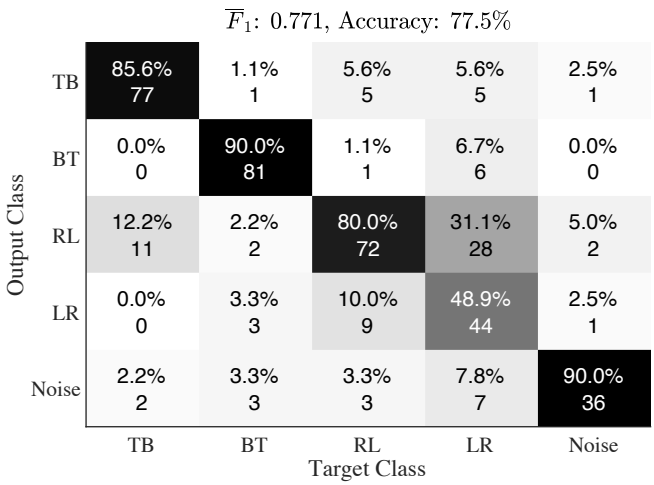


Fig. 4. Confusion matrix obtained with 5-fold cross validation and using the RBF kernel SVM with $C = 70$ and $\gamma = 0.01$.

4. CONCLUSIONS

This paper presented a novel approach for performing ultrasonic gesture recognition for handsets. The method is of low computational complexity and is based on transmitting ultrasonic pulses from the earpiece and loudspeaker and receiving the reflected signals with two microphones. SNR estimates are computed from the reflected signals on each microphone from which statistical moments are extracted and are later used to train an SVM classifier along with hyperparameter optimization. The accuracy achieved is 77.5% on a database of 400 observations. The proposed approach is robust to speaker gain changes as it models the shape of the series of SNR estimates rather than their absolute value.

5. REFERENCES

[1] GSM Association, "The mobile economy 2018," 2018.

[2] S. H. Kim and T. S. Kim, "Hand gesture recognition input system and method for a mobile phone," in *US20080089587A1*, 2011.

[3] P. Schmieder, J. Hosking, A. Luxton-Reilly, and B. Plimmer, "Thumbs Up: 3D gesture input on mobile phones using the front facing camera," in *Human-Computer Interaction – INTERACT 2013*, Berlin, Heidelberg", 2013, pp. 318–336, Springer Berlin Heidelberg.

[4] H. Lahiani, M. Elleuch, and M. Kherallah, "Real time hand gesture recognition system for android devices," in *2015 15th International Conference on Intelligent Systems Design and Applications (ISDA)*, Dec 2015, pp. 591–596.

[5] S. Gupta, D. Morris, S. Patel, and D. Tan, "SoundWave: Using the doppler effect to sense gestures," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2012, CHI '12, pp. 1911–1914, ACM.

[6] Y. Qifan, T. Hao, Z. Xuebing, L. Yin, and Z. Sanfeng, "Dolphin: Ultrasonic-based gesture recognition on smartphone platform," in *2014 IEEE 17th International Conference on Computational Science and Engineering*, Dec 2014, pp. 1461–1468.

[7] B. V. Dam, Y. Murillo, M. Li, and S. Pollin, "In-air ultrasonic 3D-touchscreen with gesture recognition using existing hardware for smart devices," in *2016 IEEE International Workshop on Signal Processing Systems (SiPS)*, Oct 2016, pp. 74–79.

[8] N. Giordano, *College Physics: Reasoning and Relationships*, Cengage Learning, 2 edition, 2009.

[9] K. Kalgaonkar and B. Raj, "One-handed gesture recognition using ultrasonic doppler sonar," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2009, pp. 1889–1892.

[10] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: Ubiquitous gesture sensing with millimeter wave radar," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 142:1–142:19, Jul 2016.

[11] M. T. I. Aumi, S. Gupta, M. Goel, E. Larson, and S. Patel, "Doplink: Using the doppler effect for multi-device interaction," in *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 2013, UbiComp '13, pp. 583–586, ACM.

[12] Ke-Yu Chen, D.l Ashbrook, M. Goel, S.H. Lee, and S. Patel, "Airlink: Sharing files between multiple devices using in-air gestures," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, New York, NY, USA, 2014, UbiComp '14, pp. 565–569, ACM.

[13] A. Das, I. Tashev, and S. Mohammed, "Ultrasound based gesture recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 406–410.

[14] Y. Sang, L. Shi, and Y. Liu, "Micro hand gesture recognition system using ultrasonic active sensing," *IEEE Access*, vol. 6, pp. 49339–49347, 2018.

[15] X. Li, H. Dai, L. Cui, and Y. Wang, "Sonic-operator: Ultrasonic gesture recognition with deep neural network on mobiles," in *2017 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computed, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Aug 2017, pp. 1–7.

[16] G. Heinzel, A. Rudiger, and R. Schilling, "Spectrum and spectral density estimation by the discrete fourier transform (DFT), including a comprehensive list of window functions and some new at-top windows," .

[17] ISO 9613-1, "Acoustics - attenuation of sound during propagation outdoors - part 1: Calculation of the absorption of sound by the atmosphere," .

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

[19] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods*, Cambridge University Press, New York, NY, USA, 2000.

[20] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, March 2002.

[21] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA, USA, 2001.

[22] S. Theodoridis, *Machine Learning: A Bayesian and Optimization Perspective*.