Sound collection and visualization system enabled participatory and opportunistic sensing approaches

Sunao Hara, Masanobu Abe
Graduate School of Natural Science and Technology
Okayama University
Okayama, Japan 700–8350
Email: hara@okayama-u.ac.jp, abe@cs.okayama-u.ac.jp

Noboru Sonehara
National Institute of Informatics
Tokyo, Japan 101-8430
Email: sonehara@nii.ac.jp

***Abstract*—This paper presents a sound collection system to visualize environmental sounds that are collected using a crowdsourcing approach. An analysis of physical features is generally used to analyze sound properties; however, human beings not only analyze but also emotionally connect to sounds. If we want to visualize the sounds according to the characteristics of the listener, we need to collect not only the raw sound, but also the subjective feelings associated with them. For this purpose, we developed a sound collection system using a crowdsourcing approach to collect physical sounds, their statistics, and subjective evaluations simultaneously. We then conducted a sound collection experiment using the developed system on ten participants. We collected 6,257 samples of equivalent loudness levels and their locations, and 516 samples of sounds and their locations. Subjective evaluations by the participants are also included in the data. Next, we tried to visualize the sound on a map. The loudness levels are visualized as a color map and the sounds are visualized as icons which indicate the sound type. Finally, we conducted a discrimination experiment on the sound to implement a function of automatic conversion from sounds to appropriate icons. The classifier is trained on the basis of the GMM-UBM (Gaussian Mixture Model and Universal Background Model) method. Experimental results show that the F-measure is 0.52 and the AUC is 0.79.**

I. Introduction

Sound is one of the most important information sources for human beings for understanding the environment around them. However, humans interpret sounds differently on the basis of their experiences and their current situation. In this study, we refer to such a sound as an environmental sound. For example, we may feel a sound is louder at night than at noon even if it is the same sound. Take another example, the cry of an infant might be felt differently by a listener who has a child, or younger sisters or brothers compared with someone who does not. Therefore, to understand environmental sounds in the real world, we need to consider contextual information, i.e., not only sound properties, but also the situation of the listener.

Many methods for sound interpretation are known, but they only provide a general interpretation of sounds. Sound properties are generally interpreted as having spectral and/or temporal parameters, such as spectrum, fundamental frequency, and loudness. On the basis of a perceptual point of view, several methods have been introduced to interpret sound properties such as critical-band analysis, octave-bandpass analysis, and A-weighting filtering. However, these methods only interpret the sound properties on the basis of a common understanding of human beings, but this is insufficient.

In this study, we developed a sound collection method applying crowdsourcing approaches in order to understand environmental sounds by considering contextual information. The sound collection is performed by one application on a smart device. The collected data fall into two types; one is user-specific and the other is statistical data. We apply two paradigms of crowdsourcing to collect the sounds; i.e., participatory sensing [1], [2] and opportunistic sensing [3]. Using the participatory sensing paradigm, we can collect sounds that participants are interested in or appreciate, therefore we applied this paradigm for collecting the raw waveform of sounds. Conversely, using the opportunistic sensing paradigm, we can collect sound statistics and, in particular, collect the loudness level as statistics in this paper.

The data should be statistically processed or anonymized to reduce any privacy risk. From this perspective, EarPhone [4] and NoiseTube [5] are important existing works. In these studies, they tried to collect environmental sounds as noise using a crowdsourcing approach; in other words, they mainly dealt with the statistics of the sound. McGraw *et al.* [6] collected sound data using Amazon Mechanical Turk as a crowdsourcing platform. Matsuyama *et al.* [7] conducted their sound-data collection using an HTML5 application and evaluated the performance of sound classifiers. Their study mainly deals with the raw waveform of sounds that cannot identify the listener. In contrast with these studies, the main contribution of our paper is to enable sound-data collection that takes contextual information into account.

We developed a visualization method for the sounds collected by participatory and opportunistic sensing paradigms. This visualization is one of the most important processes for the interpretation of environmental sounds. The waveforms of the sound are visualized by icons symbolizing the sounds at a certain location on a map, and the statistics of the sounds are visualized as colors on the same map.

Section II presents a summary of the sound collection system and Section III explains a sound collection experiment, using the system developed in Section II. Section IV presents the visualization method for sound mapping using the collected data and Section V explains an experiment to discriminate the sound type for evaluating the possibility of automatic classification of the sound collected in a real environment.

II. Development of Sound Collection System

A. Recording application for environmental sound

We developed a recording application for environmental sound. We used a Google Nexus 7, which is a 7-inch touch screen tablet for Android OS. Figures 1a and 1b show screen shots of the location logging and sound logging screens, respectively. Data recorded is starts working when the user slides the button at the upper side of the screen.

On the location logging screen, the system can record highly accurate location information using GPS, Cell-ID, or Wi-Fi networks via the Android API. The default sampling rate is one second, but this can be changed by the user through the settings. A map on the screen can show the history of the locations of the user as pin icons on the map.

On the sound logging screen, the system can record raw sound signals and calculate loudness levels using a microphone on the device. It always stores the sound data of the most recent twenty seconds using a ring buffer and also analyzes the sound to calculate the equivalent loudness level and levels of an 8-channel frequency filterbank at intervals of one second.

Annotations on the sound can be attached during the recording by users, such as subjective evaluation, sound type selection, and free description. The subjective evaluation has a five-grade scale for two metrics; one is subjective loudness level and the other is subjective crowdedness level. The sound type is easy to annotate with a selection of five preset sound types. A free description can be used as a summary of the recording environment, feelings, etc.

All of the annotations are recorded in log files with time information and a WAV file, including ten seconds of sound, is created at the same time. These can be sent to a server if the application settings permit. The sent log files are parsed on the server and they are shown in a timeline view like that of Twitter, which is shared for all users in this implementation.

(a) Location logging screen (b) Sound logging screen

Fig. 1. Screenshots of the recording application

B. Specification of the data collected by the application

The application generates sound files and three types of log file in one session. The log files are a location history log-file, loudness level log-file, and tweet log-file and they contain time information, which is triggered. In this paper, we use the *System.currentTimeMillis()* method of standard Java API for time synchronization of files because of its simplicity.

Sounds are recorded at a sampling frequency of 32000 Hz and 16-bits per second with a single channel. They are analyzed at equivalent loudness level L_{eq} per second,

$$L_{eq} = 10 \log_{10} \frac{1}{N} \sum_{n=1}^{N} (A \cdot s[n])^2 , \quad (1)$$

where $s[n]$ is a sampled signal, A is a transform factor from sampled amplitude to sound pressure level, and N is the signal length. In this paper, N is fixed to 32000, which is equivalent to 1 s. A is detected by a preliminary examination to compare with values of a sound level meter, RION NL–42. The lower limit of loudness level by the application on the Nexus 7 is approximately 40 dB(FLAT) because of the performance of the device's A/D equipment.

Addition to L_{eq}, this system can also record filter bank output levels in 8-channel, which is related to octave band filter analysis. The filter is implemented using triangle windows. The central frequencies of the filter are $f_c = [63, 125, 250, 500, 1000, 2000, 4000, 8000]$.

C. Server application for collection and exploration of the sounds

The client and server applications communicate via HTTP protocols. The server implements APIs for receiving and browsing data and the browsing API can create not only a general HTML view for WWW browsers, but also a JSON (JavaScript Object Notation) view for advanced applications.

The server system includes several open-source softwares. The server OS is a Debian GNU/Linux 7.5 (Wheezy) as a virtual machine on VMware ESXi 5.1. The web application framework is Mojolicious [1] with Perl language. The back-end database software is MongoDB [2]. The application is running on Mojo::Server::Hyptonotoad [3] with an nginx front-end server [4]. The system will be used for the crowd-sourced sound recordings, hence, a large number of users will use the system, and hence it must have the appropriate processing capacity. These software have a distributed computing architecture that might be an answer to the problems of heavy usage.

III. Loudness and Environmental Sounds Data Collection

A. Conditions of data collection

A data collection experiment was conducted on ten participants comprising two faculty members and eight graduate students, who commute to Okayama University to work. The

[1] http://mojolicio.us/
[2] http://www.mongodb.org/
[3] http://mojolicio.us/perldoc/Mojo/Server/Hypnotoad
[4] http://nginx.org/

participants were instructed how to use smart devices and the data collection applications. They were asked to collect the sounds, annotations, and loudness levels during travel to and from work. Some were also asked to collect data near their home or railway stations.

They recorded loudness levels during application running and the sound with the annotations at various intervals by user. Data collection was expected to be conducted by the participants holding the devices in their hands during data collection because of the UI design of the client application, which is an appropriate condition for collecting clear sound. However, footstep noise could be mixed in with the recorded sound because participants might be handling the device during walking, which may cause a biased value in the loudness levels.

The subjective loudness level is evaluated in five scales: L_1: very quiet, L_2: relatively quiet, L_3: relatively noisy, L_4: quite noisy, and L_5: very noisy. The subjective crowdedness level is also evaluated in five scales: S_1: empty, S_2: sparse, S_3: relatively crowded, S_4: quite crowded, and S_5: crowded. The choices of subjective evaluations are recorded as a part of annotations.

The sound file that contains the last ten seconds of sound is created by pushing the tweet button on the sound logging screen (Fig. 1b). To add an annotation to the sound, participants selected the sound type before pushing the tweet button. Five types of sounds are preset for ease of use and users are allowed to select multiple choices. The choices are T_1: human speech, T_2: sound of birds, T_3: sound of insects, T_4: sound of cars, and T_5: sound of wind. Additionally, they can input free text for annotating the sound or recording environment. They are not required to fill in all of the selections, hence, they can input just one part with an annotation if they want to check one or more metrics.

B. Preliminary analysis of collected data

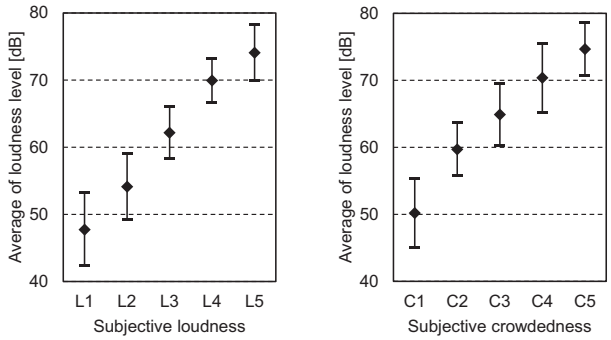
Data was collected mainly for areas near Okayama University, including neighboring residential estates, and at an Okayama railway station.

All of the collected data were synchronized with their time information, and we obtained 6,257 loudness data with a tuple of latitude, longitude, and time. The sound data comprised 516 collected samples with ten seconds of sounds with the same tuples. A distribution of the sound data collected for each type is shown in Table I.

Figure 2a and 2b are the average loudness levels as functions of the subjective loudness level and subjective crowdedness level, respectively. The average value is calculated as the average of the data from the ten participants, and the error bars are indicative of 90%-confidential intervals. We find two classes for subjective loudness to consider overlapping error bars in Figure 2a, that is, L1, L2, and L3 are the "quiet" class, and L4 and L5 are the "noisy" class. Figure 2b has a similar tendency, that is, L1 is "empty" and the others are "not empty." The existence of two classes is not trivial because of the design of our questionnaires; however, its long error bars display the importance of listener-specific information for sound interpretation.

TABLE I. TYPE OF ENVIRONMENTAL SOUND AND ITS DISTRIBUTION

	Type of sound	unlabeled (-1)	labeled ($+1$)
T_1	Human speech	386	130
T_2	Sound of birds	447	69
T_3	Sound of insects	479	37
T_4	Sound of cars	318	198
T_5	Sound of wind	492	24



(a) Subjective loudness (b) Subjective crowdedness

Fig. 2. Average Loudness as a function of a subjective evaluation. The error bars indicates 90% confidence intervals as estimations of average loudness levels for the subjective level of each participant.

IV. VISUALIZATION SYSTEM FOR LOUDNESS AND ENVIRONMENTAL SOUNDS

The system statistically processes loudness data with spatio-temporal indexes; latitude, longitude, and time. The amount of data, sum of data, and squared-sum of data in each index is calculated as a sufficient statistic of Gaussian distribution. These parameters are updated on demand by uploading data from users.

Visualization of the loudness data is a color map of each area. The color index is calculated from the average of loudness. We can overview the loudness distribution of any interesting district on the map. An example of the visualization of loudness levels is shown in Fig. 3. The color indicates the average of the loudness level, so for example, red means a higher loudness than blue. The transparency shows the amount of data in the area, so for example, the weaker the transparency is, the less the amount of data. In other words, weak transparency means the data is not confident.

As mentioned in the Introduction, the experience and the situation of the listner affect how to feel the sounds. The Fig. 3 tells us that the area near the university is more quiet than the area near the railway station in the point of view of loudness level. However, it is just interpreted the sound properties on the basis of common understanding of human beings as usual way. If you know that the area near the university is obtaining its lecture buildings and apartments for the college students, you may call the area as "quiet district" rather than "deserted district." Similarly, the area near the railway station is downtown area, therefore, the area should call as "bustle and busy district" rather than "noisy district."

Sound visualization is realized by icons symbolizing the sound on the map so we can see the sound types in any district that interests us. An example of environmental sound visualization is shown in Fig. 4. The sounds are distinguished by icons on the basis of their subjective evaluations during

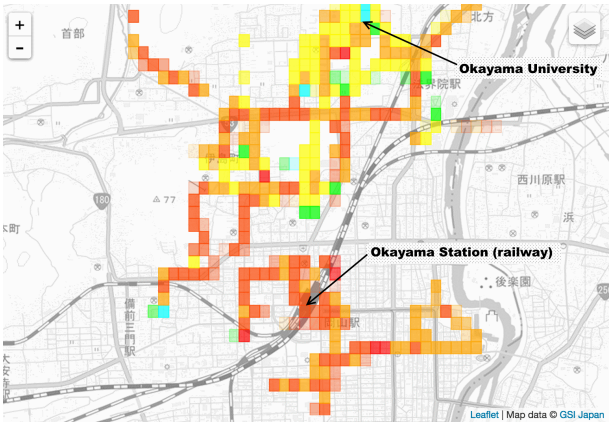


Fig. 3. Sound map visualizing loudness level by color

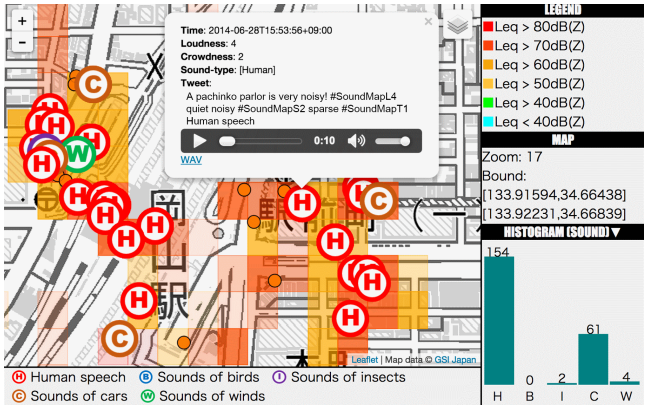


Fig. 4. Sound map visualizing sound type by icons

recording. The icons can be clicked to browse the sound's information and listen to it. The right side of the map interface shows the histogram of the sound type as statistics in the current viewing area.

V. Experiment in Environmental Sound Discrimination

An experiment in environmental sound discrimination was carried out. Five classifiers were created for the five types of sounds described in Table I and these are used for visualization of the sound on the map. The current visualization of sound type is based on manual annotations by the users, as described in Section IV. If automatic discrimination is realized, the system can automatically convert the sound to an icon. Automation is a very important factor in crowdsourcing.

We evaluated through four-fold cross validations to assume a subject-opened condition. In this experiment, 516 samples from 10 participants were collected. Moreover, data cleaning was performed by excluding two participants because they collected less data; so there were 514 samples from 8 participants. Finally, we clustered the participants into four clusters with as nearly equal amounts of data as possible. We test the data from participants in a certain cluster using the model which is trained by the data from participants in another three clusters, then we repeat the trials with changing the test data four times.

A. Feature extraction

We used Mel-Frequency Cepstral Coefficient (MFCC) as an acoustic feature, which is commonly used for speech recognition or speaker recognition. We extracted 50 dimensional features by MFCC, on the basis of an existing study [8]. First, sounds were analyzed by a Hamming window that has parameters of 25-ms window length and 10-ms window shift size. Next, the window-processed sounds were analyzed by a 40-channel filterbank that had a bandwidth between 30 and 16000 Hz and then converted to MFCCs. Finally, we used 16-dimensional MFCCs and their first-order and second-order derivatives. Note that we excluded the 0th MFCC but included first-/second-order derivatives of the 0th MFCC because it indicates the energy of the sound. We called this feature as `MFCC50`.

We compared our results with general automatic speech recognition parameters. One comprised a 12-dimensional MFCC and the first derivative of the MFCC and energy, called `MFCC25`. The other comprised a 12-dimensional MFCC and the first-/second-order derivatives of the MFCCs and energy, called `MFCC38`. Note that these parameters were analyzed in a bandwidth between 0 to 16000 Hz. An experimental condition of feature extraction is shown in Table II.

B. Discrimination method based on GMM-UBM

The discrimination method was based on a GMM-UBM [9], which is generally used for speaker recognition. GMM is a kind of generative models and we assume that the model can represent the properties of the environmental sounds. A likelifood $\mathcal{L}$ of the sound for the GMM $\mathcal{M}$ is calculated as below:

$$\mathcal{L}(\boldsymbol{o};\mathcal{M}) = \sum_m \lambda_m f(\boldsymbol{o};\boldsymbol{\mu},\boldsymbol{\sigma}), \tag{2}$$

$$f(\boldsymbol{o};\boldsymbol{\mu},\boldsymbol{\sigma}) = \frac{1}{\sqrt{2\pi\boldsymbol{\sigma}^2}} \exp\left(-\frac{(\boldsymbol{o}-\boldsymbol{\mu})^2}{\boldsymbol{\sigma}^2}\right), \tag{3}$$

where $\boldsymbol{o}$ is a feature vector of the sound, $\mathcal{M}$ is GMM training from a training data, λ_m is mixture weight of the components, and a covariance matrix of the GMM was assumed as a diagonal matrix.

In general, GMM is trained for categorizing sound by EM algorithm. The EM algorithm is a kind of iterative learning method. The mixture components of GMM are representing the variability of training data. On the other hands, UBM $\mathcal{M}_{ubm}$ is also GMM, but it trained for all non-categorized sound. The UBM contains a lot of mixture components, therefore, its components modeling the average properties of each categorized environmental sound. In other words, we can assume that all important mixture components of each

TABLE II. Experimental Condition of Feature Extraction

	Bandwidth	Filterbank	Feature components
`MFCC25`	0–16000 Hz	25 channels	MFCC (1st–12th) + ΔMFCC (0th–12th)
`MFCC38`	0–16000 Hz	25 channels	MFCC (1st–12th) + ΔMFCC (0th–12th) + ΔΔMFCC (0th–12th)
`MFCC50`	30–16000 Hz	40 channels	MFCC (1st–16th) + ΔMFCC (0th–16th) + ΔΔMFCC (0th–16th)

class model are contained in the UBM. If we adapt or train class model using UBM as the seed model, we can obtain the coordinated mixture components between class model and UBM model. It is an important property to introduce the Log likelihood ratio (LLR).

First, the UBM model is trained for all data to construct the classifier. Then, the UBM is used as a seed model in class models $\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3, \mathcal{M}_4$, and $\mathcal{M}_5$, corresponding to classes $T_1, T_2, \cdots, T_5$, respectively. The class models are created on the basis of MAP (Maximum a posteriori) adaptation method [10] for each class. We use a hyper-parameter $\tau = 10$.

A retraining model is also created for comparison. The retraining model is expected to achieve higher accuracy than the MAP adaptation model if the amount of training data is sufficiently large.

The final decision is performed by thresholding a log likelihood ratio (LLR) calculated by the UBM and each class model. This classifier can detect whether a certain class of sound is included in the test sound $(+1)$ or not (-1). The LLR of class T_k for the test sound $\boldsymbol{o}$ is calculated from the likelihood function $\mathcal{L}(\boldsymbol{o}; \mathcal{M})$:

$$LLR(\boldsymbol{o}, T_k) = \log \frac{\mathcal{L}(\boldsymbol{o}; \mathcal{M}_k)}{\mathcal{L}(\boldsymbol{o}; \mathcal{M}_{ubm})}. \tag{4}$$

Then, the classifier $\mathcal{C}_k(\cdot)$ of class k is constructed as the following equation:

$$\mathcal{C}_k(\boldsymbol{o}) = \begin{cases} +1 & \text{if} \quad LLR(\boldsymbol{o}, T_k) > 0, \\ -1 & \text{if} \quad LLR(\boldsymbol{o}, T_k) \leq 0. \end{cases} \tag{5}$$

The mixture number of the GMM is updated by twice the number; i.e., training a 2-mixture model from the 1-mixture model, a 4-mixture model from the 2-mixture model, $\cdots$, and finally training a 256-mixture model from the 128-mixture model. EM training is repeated for a maximum of 20 times for each mixture number. The training and evaluation toolkit is HTK 3.4.1 [10].

C. Evaluation by F-measure

The F-measure is calculated from the following equation

$$F = \frac{2N_{tp}}{2N_{tp} + N_{fn} + N_{fp}}, \tag{6}$$

where N_{tp} is the number of true positives, N_{fp} is the number of false positives, N_{fn} is the number of false negatives, and N_{tn} is the number of true negatives. In general, the appropriate number of GMM mixture components is different between classes. Therefore, we used the result produced by the model that achieved the maximum F-measure value, instead of preselecting the number of mixtures.

Figure 5 shows the average F-measure for each class. The highest F-measure of $F = 0.522$ is achieved when using the 25-dimensional features with the retraining method. The retraining model is always more accurate than the MAP adaptation method.

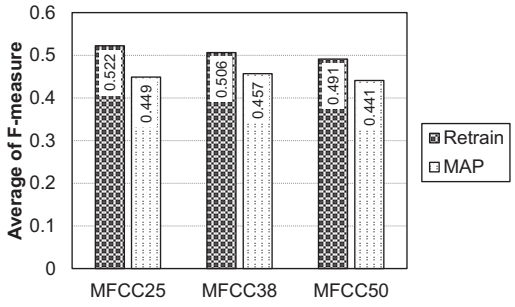


Fig. 5. The average F-measure of five classifiers

D. Evaluation by ROC curve

We add threshold parameter θ to the classifier $\mathcal{C}_k(\cdot)$, and called it $\mathcal{C}_{k,\theta}(\cdot)$, to create ROC (Receiver Operating Characteristic) curves.

$$\mathcal{C}_{k,\theta}(\boldsymbol{o}) = \begin{cases} +1 & \text{if} \quad LLR(\boldsymbol{o}, T_k) > \theta, \\ -1 & \text{if} \quad LLR(\boldsymbol{o}, T_k) \leq \theta. \end{cases} \tag{7}$$

For example, an ROC curve of classifier "sound of birds (T_2)" is shown in Fig. 6. This figure indicates that a more appropriate threshold exists to achieve a higher F-measure than the value involved in the previous section.

The average AUC (Area Under the Curve) for each classifier is shown in Fig. 7. This figure shows similar F-measure results to those in Fig. 5. The highest performance of 0.795 is achieved by the retraining model with the 25-dimensional features. However, the difference between features and training methods is negligible. Therefore, the difference in accuracy is very small between the choice of models for use if the appropriate threshold is given.

VI. Conclusion

In this paper, we developed a server-client application for collection of environmental sounds using smart devices that are enabled for participatory and opportunistic sensing approaches. We conducted a sound collection experiment with ten participants using the developed application. The collected data are analyzed for the distribution of loudness levels and sound type, and they are visualized on a map as a color map and icons, respectively. We then conducted a discrimination experiment to evaluate the performance in discriminating the existence of target classes of sounds for five classes.

The effectiveness of the application has been demonstrated through the experiments, but there remains some future work that can be done. For example, the microphone specification should be appropriately calibrated if it is to be used in a real crowdsourcing environment. For this purpose, we can examine the calibration information of different devices. Speaker adaptation as a method is promising, i.e., the calibration parameters of an unknown device is calculated from the parameters of known devices and a small amount of data recorded by the unknown device. We have approached sound recordings as a participatory sensing paradigm; however, analyzing sound recordings through opportunistic sensing paradigm might provide more information. To enable an opportunistic sensing approach, a privacy protection method must be developed.

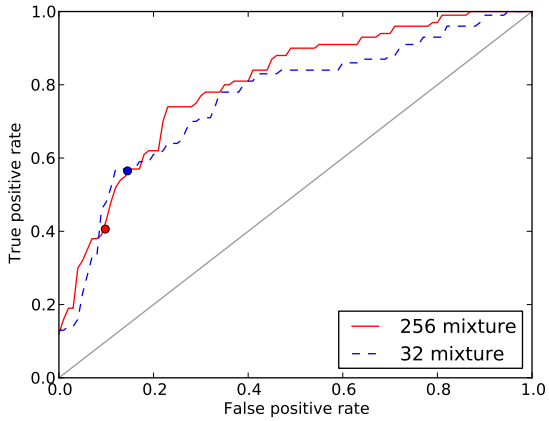


Fig. 6. Detection accuracy using the 50-dimensional features. A circle mark means a result of $\theta = 0$.

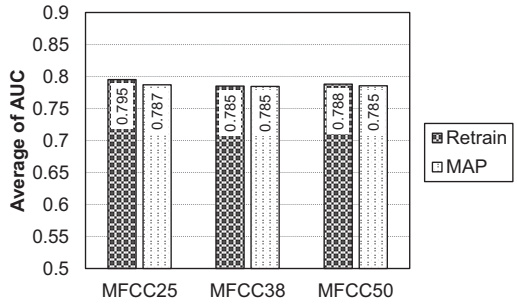


Fig. 7. Detection accuracy by AUC

REFERENCES

[1] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory sensing," in *Proceedings of ACM workshop of World-Sensor-Web*, ser. ACM Sensys, Oct. 2006, pp. 117–134.

[2] J. Goldman, K. Shilton, J. A. Burke, D. Estrin, M. Hansen, N. Ramanathan, S. Reddy, V. Samanta, M. Srivastava, and R. West, "Participatory sensing: A citizen-powered approach to illuminating the patterns that shape our world," Woodrow Wilson International Center for Scholars, Washington, D.C., May 2009.

[3] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson, "People-centric urban sensing," in *Proceedings of the 2nd Annual International Workshop on Wireless Internet*. Article No. 18, ACM, 2006.

[4] R. Rana, C. Chou, S. Kanhere, N. Bulusu, and W. Hu, "Ear-Phone: An end-to-end participatory urban noise mapping system," in *Proceedings of The 9th ACM/IEEE International Confernce an Information Proccessings in Sensor Networks (IPSN 2010)*, Apr. 2010, pp. 105–116.

[5] E. D'Hondt, M. A. Stevens, and A. Jacobs, "Participatory noise mapping works! an evaluation of participatory sensing as an alternative to standard techniques for environmental monitoring," *Pervasive and Mobile Computing*, vol. 9, no. 5, pp. 681–694, Oct. 2013.

[6] I. McGraw, C.-y. Lee, L. Hetherington, S. Seneff, and J. R. Glass, "Collecting voices from the cloud," in *Proceedings of LREC 2010*, May 2010, pp. 1576–1583.

[7] M. Matsuyama, R. Nisimura, H. Kawahara, J. Yamada, and T. Irino, "Development of a mobile application for crowdsourcing the data collection of environmental sounds," in *Human Interface and the Management of Information. Information and Knowledge Design and Evaluation*, S. Yamamoto, Ed. Springer, 2014, vol. 8521, pp. 514–524.

[8] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *Proceedings of 18th European Signal Processing Conference (EUSIPCO-2010)*, Aug. 2010, pp. 1267–1271.

[9] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[10] "The HTK Book," `http://htk.eng.cam.ac.uk/`.