Advances in Science, Technology and Engineering Systems Journal Vol. 2, No. 6, 145-150 (2017)

www.astesj.com

Special issue on Advancement in Engineering Technology

ASTESJ

ISSN: 2415-6698

A Relational Database Model and Tools for Environmental Sound Recognition

Yuksel Arslan[*], Abdussamet Tanıs, Huseyin Canbolat

Electrical and Electronics Engineering, Ankara Yıldırım Beyazıt University, 06010, Turkey

A R T I C L E I N F O

Article history:
Received: 29 October, 2017
Accepted: 17 November,2017
Online: 10 December, 2017

Keywords:
Database Design
Environmental Sound Recognition
Machine Learning

A B S T R A C T

Environmental sound recognition (ESR) has become a hot topic in recent years. ESR is mainly based on machine learning (ML) and ML algorithms require first a training database. This database must comprise the sounds to be recognized and other related sounds. An ESR system needs the database during training, testing and in the production stage. In this paper, we present the design and pilot establishment of a database which will assists all researchers who want to establish an ESR system. This database employs relational database model which is not used for this task before. We explain in this paper design and implementation details of the database, data collection and load process. Besides we explain the tools and developed graphical user interface for a desktop application and for the WEB.

1. Introduction

This paper is an extension of work presented in 25th Signal Processing and Communications Applications Conference (SIU), 2017 [1]. The database design and implementation described in that paper was mainly for impulsive sound detection and the database was for a hazardous sound recognition application. Here the extended database is for all kinds of environmental sounds and it is aimed for all kinds of ESR applications.

Historically non-speech sound recognition has not received as much attention as automatic speech recognition (ASR). ASR has well established algorithms and databases while research has begun much earlier. Automatic ESR (AESR) is getting attraction since last two decades. We can list the following applications of AESR: In military, forensic and law enforcement domain there are studies on gunshot detection systems. In [2], a gunshot detection system is proposed. In [3], the gunshot blast is used to identify the caliber of the gun. In [4] and [5] ESR is used for robot navigation. ESR can be used for home monitoring. It can be used to assist elderly people living in their home alone [6], [7]. In [8], it is used for home automation. In [9] and [10], ESR is used for recognition of animal sounds. In the surveillance area, it is used for surveillance of road [11], public transport [12], elevator [13] and office corridor [14].designations. ESR system design is started with the training phase. How a sound database is used in training phase of an ESR system is explained in Figure 1.

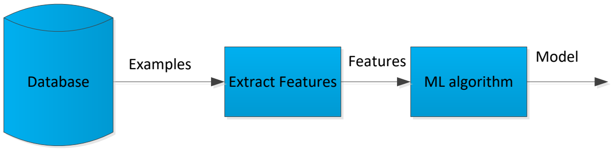


Figure 1: ESR system in training phase

During the development stage of an ESR system the desired ML algorithm must be trained with the sounds to be recognized. The database provides the sound clips to be recognized and other sound clips which are negative examples. After training, a model is developed and this model is used for testing. In Figure 1, the database provides positive and negative examples; features are inputs to the ML algorithm, ML algorithm using these features produce the model.

In Figure 2, testing phase of an ESR system is shown. In testing phase database provides the positive and negative examples, model produces the predictions about the examples provided and at last predictions are compared with the truth provided by the database and a performance rate is reached. According to this performance rate, ML algorithm or the feature set or other parameters of the ML algorithm may be needed to change. Then the whole training and testing phase start again. This process continues until an acceptable performance rate is reached.

After model creation and testing, this model is used in the production phase (Figure 3). In production phase, sounds come

[*]Yuksel Arslan, Ankara, Cankaya Ilkbahar Mah. Guneypark Kumeevleri A35/35 Turkey, Email: yuksel_arslan@yahoo.com

https://dx.doi.org/10.25046/aj020618

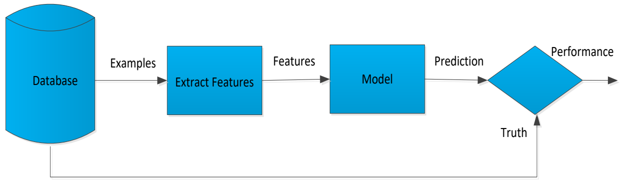


Figure 2: ESR system in testing phase

from the environment, model makes the predictions about the classes of sounds and we actually may not know the real truth. In production stage we may update sounds in database, update the model even we may change the ML algorithm. In each case we need the database.

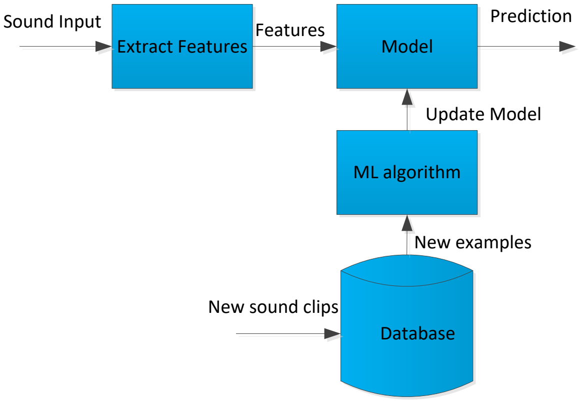


Figure 3: ESR system in production phase

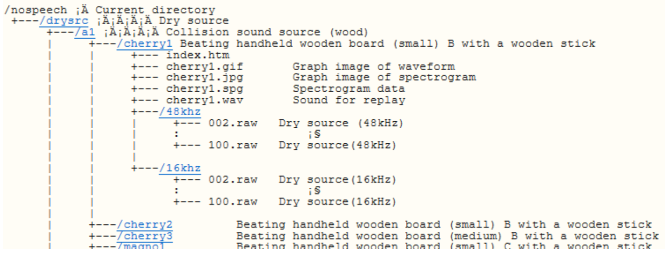
This paper is organized as follows: In section II we will make a literature review of ESR databases. In section III a new relational database model for ESR systems will be explained. At the last, section IV, the contribution of this work and other planned activities will be explained.

2. Literature Review

In this section we will review the mostly referenced databases in ESR related papers. The structure of the databases will be explored; pros and cons of the structure will be argued if applicable.

One of the databases that have been mostly reverenced is Real World Computing Partnership's (RWCP) non-speech database [15]. Using a standard single microphone, acoustic signals of about 100 types of sound sources were measured in an anechoic room as the dry source. By using the impulse response of three kinds of rooms, convolution with the 48 kHz sampling signal was carried out to reconstruct the sound signal in respective rooms. The database structure (structure of directories) and navigation to sounds is explained and provided by an HTML page as shown in Figure 4 for dry sources. As an ESR developer we must examine the structure of HTML file and find the desired sound clips. Maintenance of the data and usage is not so easy with this kind of structure.

Detection and Classification of Acoustic Scenes and Events (DCASE) is an official IEEE Audio and Acousti Acoustic Signal Processing challenge. For challenge a database is prepared and it is also a resource for ESR researchers. DCASE 2016 challenge consists of four tasks [16]. The goal of Task-2 is to detect sound events for example "bird singing", "car passing by" that are present within an audio segment. To be prepared for the challenge two datasets, train and development are given. Train is used to create the model and development is to test the model.

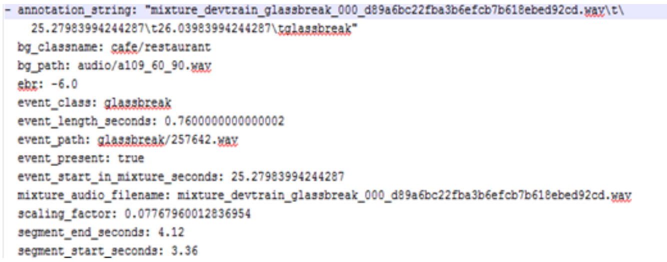


```
/nospeech ¡Ä Current directory
 +---/drysrc ¡Ä¡Ä¡Ä¡Ä Dry source
      +---/a1 ¡Ä¡Ä¡Ä¡Ä Collision sound source (wood)
      |     +---/cherry1 Beating handheld wooden board (small) B with a wooden stick
      |     |     +--- index.htm
      |     |     +--- cherry1.gif       Graph image of waveform
      |     |     +--- cherry1.jpg       Graph image of spectrogram
      |     |     +--- cherry1.spg       Spectrogram data
      |     |     +--- cherry1.wav       Sound for replay
      |     |     +---/48khz
      |     |     |     +--- 002.raw   Dry source (48kHz)
      |     |     |     :                  ¡§
      |     |     |     +--- 100.raw   Dry source(48kHz)
      |     |     |
      |     |     +---/16khz
      |     |           +--- 002.raw   Dry source(16kHz)
      |     |           :                  ¡§
      |     |           +--- 100.raw   Dry source(16kHz)
      |     |
      |     +---/cherry2          Beating handheld wooden board (small) B with a wooden stick
      |     +---/cherry3          Beating handheld wooden board (medium) B with a wooden stick
      |     +---/magno1           Beating handheld wooden board (small) C with a wooden stick
```

Figure 4: RWCP dry source sound clips structure

For training all sound files are in one directory and a readme file explains the details, such as sampling frequency, quantization bit depth, etc. For development dataset there is an annotation text file for each sound clip. Each .txt file contains information about the onset, offset, and event class for each event in the scene, separated by a tab.

DCASE 2017 Task-2 dataset contains ".yaml" files for annotations [17]. ".yaml" files (Figure 5) can be read by a Matlab command, so it is easy to work with these structured files.

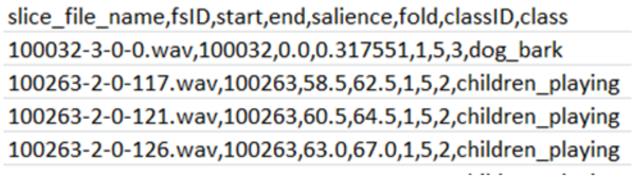


```
- annotation_string: "mixture_devtrain_glassbreak_000_d89a6bc22fba3b6efcb7b618ebed92cd.wav\t\
    25.27983994244287\t26.03983994244287\tglassbreak"
  bg_classname: cafe/restaurant
  bg_path: audio/a109_60_90.wav
  ebr: -6.0
  event_class: glassbreak
  event_length_seconds: 0.7600000000000002
  event_path: glassbreak/257642.wav
  event_present: true
  event_start_in_mixture_seconds: 25.27983994244287
  mixture_audio_filename: mixture_devtrain_glassbreak_000_d89a6bc22fba3b6efcb7b618ebed92cd.wav
  scaling_factor: 0.07767960012836954
  segment_end_seconds: 4.12
  segment_start_seconds: 3.36
```

Figure 5 DCASE 2017 task-2 development dataset "yaml" file for glass break

Dataset for Environmental Sound Classification (ESC) contains two databases. There are 10 classes and each class has 40 clips in ESC-10 dataset. ESC-10 is subset of ESC-50 which contains 50 classes and each class has 40 clips. There is a readme file in which there is a line for each clip explaining the details of the clip for each dataset [18].

In [19], a dataset of annotated urban sound clips are recorded and taxonomy for urban sound sources is proposed. The dataset contains 10 sound classes with 18.5 hours of annotated sound event occurrences. The dataset contains a CSV file (Figure 6) explaining the details of each recording.



```
slice_file_name,fsID,start,end,salience,fold,classID,class
100032-3-0-0.wav,100032,0.0,0.317551,1,5,3,dog_bark
100263-2-0-117.wav,100263,58.5,62.5,1,5,2,children_playing
100263-2-0-121.wav,100263,60.5,64.5,1,5,2,children_playing
100263-2-0-126.wav,100263,63.0,67.0,1,5,2,children_playing
```

Figure 6: CSV file explaining Urbansound database

Another ESR database is http://www.desra.org. In [20], the aim of this database, details of the design and the sources used are explained. It is designed as multi-purpose database. The database contains variety of sounds from different events with thorough background information for each sound file. It is accessible from the Internet (The database is not fully functional at the moment). The database was designed considering for admin tasks and general user level tasks. Web front end provides the functionality for user level tasks.

http://www.auditorylab.org is [21] another database. This database was constructed by Carnegie Mellon University to examine the human ability to use sounds to understand what events are happening in the environment. All the sounds in the database are recorded in a controlled way. The laboratory and the recording media used are technically detailed. In the construction of database sounds are grouped by the event that makes the sound. Sounds of events like impact, rolling can be downloaded from this database.

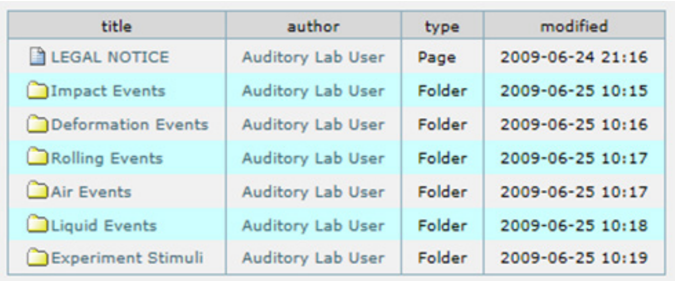
3. Database Design

The databases reviewed in Section 2 are prepared for just resources for the development of ESR applications. Many of them provide the data and the files for correctly handling the data. In this context they are valuable resources for all ESR researchers. Desra [20], provides extra tools such as a web graphical user interface for searching and testing.

In the development process of an ESR system, some main functions training, testing and production explained in Section I. During this development process, many sound clips are used; new sound clips may be added or deleted. We deal with many features extracted from these sound clips; we create models using different ML algorithms. Then we compare the models; try to find best features and best ML algorithms. This loop continues. Our first goal in this database design is to help researchers as much as possible to ease the burden of handling data. The second goal is to help maintenance of the data. Our data is sound clips, features, algorithms and models. Addition, deletion, searching, annotation, backup and restore can be thought as the maintenance task.

3.1 Taxonomy of Environmental Sounds

We need a taxonomy to be able to store, search and retrieve the data from the database. During literature review we see two kinds of taxonomy. In the first taxonomy, the sounds can be grouped by the event that makes the sound. In Figure 7, grouping of sounds defined in [21] is seen.



title	author	type	modified
LEGAL NOTICE	Auditory Lab User	Page	2009-06-24 21:16
Impact Events	Auditory Lab User	Folder	2009-06-25 10:15
Deformation Events	Auditory Lab User	Folder	2009-06-25 10:16
Rolling Events	Auditory Lab User	Folder	2009-06-25 10:17
Air Events	Auditory Lab User	Folder	2009-06-25 10:17
Liquid Events	Auditory Lab User	Folder	2009-06-25 10:18
Experiment Stimuli	Auditory Lab User	Folder	2009-06-25 10:19

Figure 7: Grouping of sounds based on sound producing events [21]

Another hierarchical grouping is seen in Figure 8. This is also classified in the first taxonomy. This is the grouping defined in [22] based on sound producing events and the listeners' description of the sound. Second taxonomy is based on the sound source [19][23]. The subset of the taxonomy defined in [23] is taken into consideration for urban sounds given in [19].

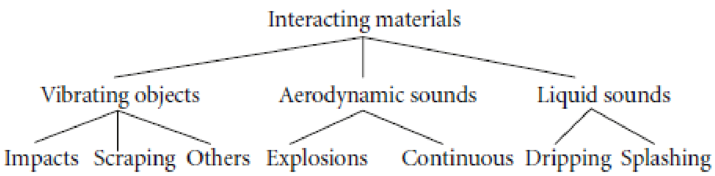


Figure 8: Grouping of sounds defined in [22]

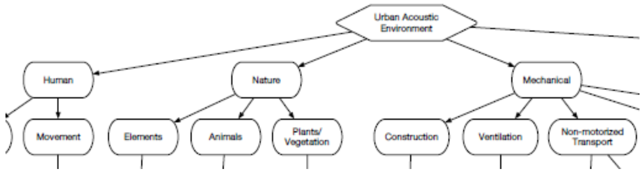


Figure 9 A part of the urban sound taxonomy defined in [19]

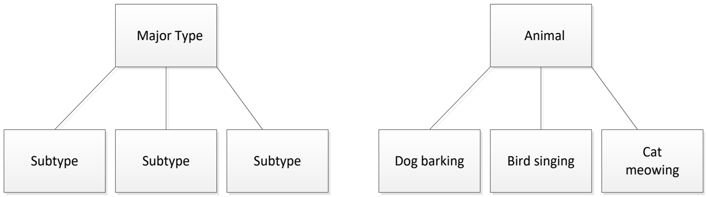


Figure 10: Environmental sounds grouping in the database

3.2 Non Functional Requirements of the Database

- There will be sound clips in the database. These sound clips will contain environmental sounds. Each sound clip can have more than one environmental sound. These environmental sound clips can belong to different major environmental sound types. Origin of these sound clips, such as own recording, from an internet site or from another database, should be entered to database. Recording details should be entered. Size, file type of sound clip and the path where the sound clips are recorded should be entered to the database.

- There should be major types, such as human, nature, animal, etc. There should be subtypes, such as dog bark, gunshot etc. and these must belong to major type. Start and end sample index or start/end time of these clips should be known.

- There will be environmental sound clips in the database. These clips should be extracted from sound clips defined in first paragraph. Each of these sound clips should have a subtype and it should be known from which sound clips it is extracted. Extraction method should be entered. If it is extracted by an algorithm the algorithm name otherwise as "manual" should be entered to database.

- Recording details should be entered to the database such as, sampling rate, quantization bit, channel size, etc.
- Background clips should be entered to the database. Each background clip should have a type. Each background clip should also have file type, file size, recording detail and the path where it is recorded.

3.3 Functional Requirements

- There should be scripts which will take algorithms as arguments and extract the environmental sounds from sound clips. These extracted clips should be entered to the database as explained in 3.2.
- There should be scripts which will embed environmental sound into noise clips at desired Signal to Noise Ratio (SNR) level.
- There should be scripts which will extract features from environmental sound clips and store to the database.
- The scripts, their help files, paths should be stored in the database.

3.4 Graphical User Interface (GUI) Requirements

- Administrators can use the GUI for meta data entrance, deletion and update such as major types, subtypes, recording details.
- Administrators can use the GUI for data load manually or using the data loader script.
- Researchers can use the GUI for searching and downloading the desired environmental sounds.
- Researchers can load their features and models to the database.

3.5 Database Implementation

The implemented database will include some data which can be stored by way of data types found in the standard database software and also sound files. These sound files will not be stored to database instead they will be stored in the operating system file structure and the path to this file is just recorded in the database.

Microsoft SQL 2008 Database Server is used to create tables. The tables and the relationships between them are shown in Figure 11. The scripts providing functionality are coded in Matlab 2011a. The files are stored in "mat" type when required.

For functional requirements the following scripts are coded using Matlab.

- Environmental Sound Detector: This script is an interface between algorithms that extracts environmental sounds from the sound clips. Different algorithms can be used here for extraction of the environmental sounds. The algorithms must conform to this script interface definition. This script takes the algorithm name that will be started and the environmental sound types which will be searched are given as arguments.
- Environmental Sound Embedder: This script takes the type of the environmental sound, noise type, SNR level and at last the number of required record count to be created. The script merges the environmental sound clip with the noise clip at the desired SNR level and records it to the database table.
- Feature Extractor: This script acts as an interface between feature extractors. It takes the path of the feature extractor from the features table, feature name and the environmental sound type from command line. After extracting features, it is saved as a mat file.
- Data Loader: To load data from other databases this script is used to interface with data loading scripts.

The GUI is developed using Microsoft Visual Studio with C# language. GUI has admin utilities and end user tools. In Figure 12(a), it is seen the part of GUI providing admin operations. These admin operations are additions and deletions of major types, subtypes and sampling information. Besides the GUI provides the administrators load data one by one or using a script to load as a batch.

The GUI provides the users some tools. The tools are for searching the database and testing their algorithms on the sound clips given by the database. Users can search the database to see the sound clips with desired type.

End users can search the database with noise clip type, sound clip type and with SNR interval then select a clip and copy it to their own computer. After finding the start index of the embedded environment sound clip, by writing the start index of this environmental sound to the edit box on the GUI and by clicking the test button, they can test their algorithm correctness. The WEB GUI for end user tools is seen in Figure 12(b).

After database and scripts implementation we loaded the data from Urbansound [19] database. Now many operations on the database can be done either using GUI or an SQL command line.

4. Conclusion

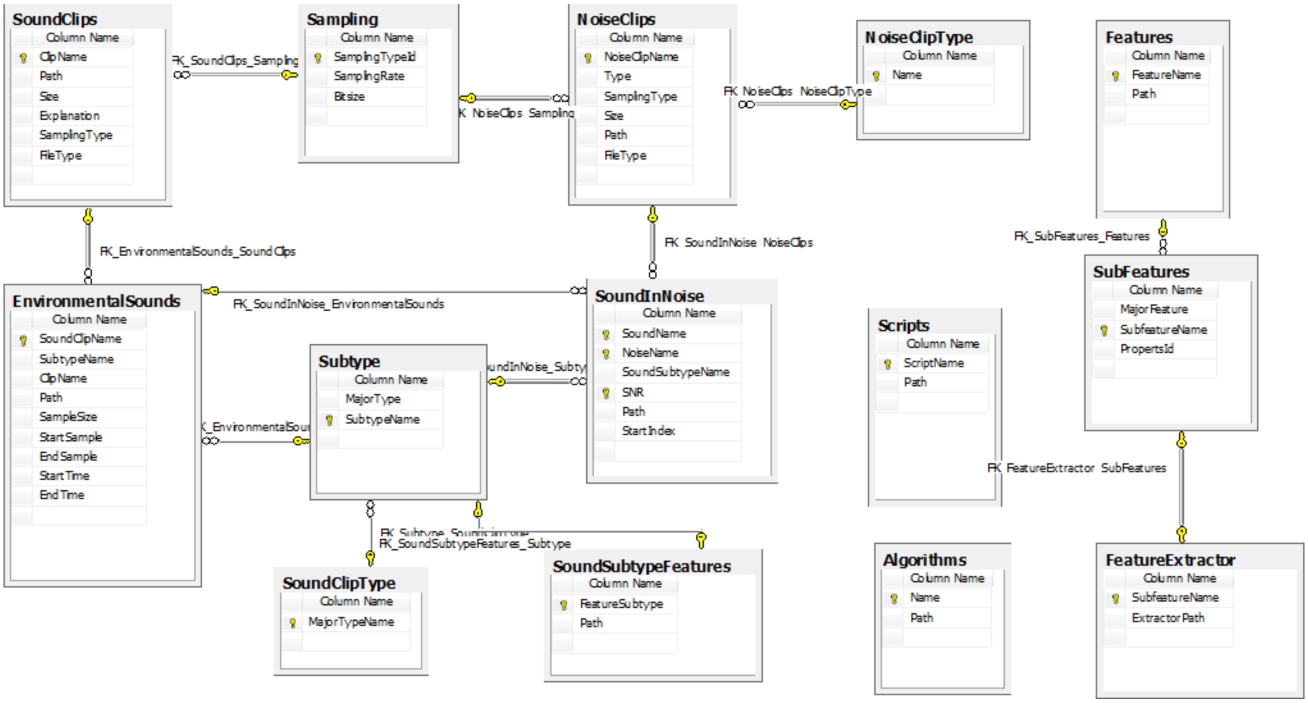
The lack of common database for environmental sound recognition is an important obstacle in front of the researchers. The development of and ESR system is a tough process during which someone have to deal with lots of sound clips with different types, algorithms and models. In this paper we explained a relational database model which will make the data handling easier. The database developed is different from other counterparts which are just providing the data. The relational database model described here provides easy maintenance as well as easy usage.

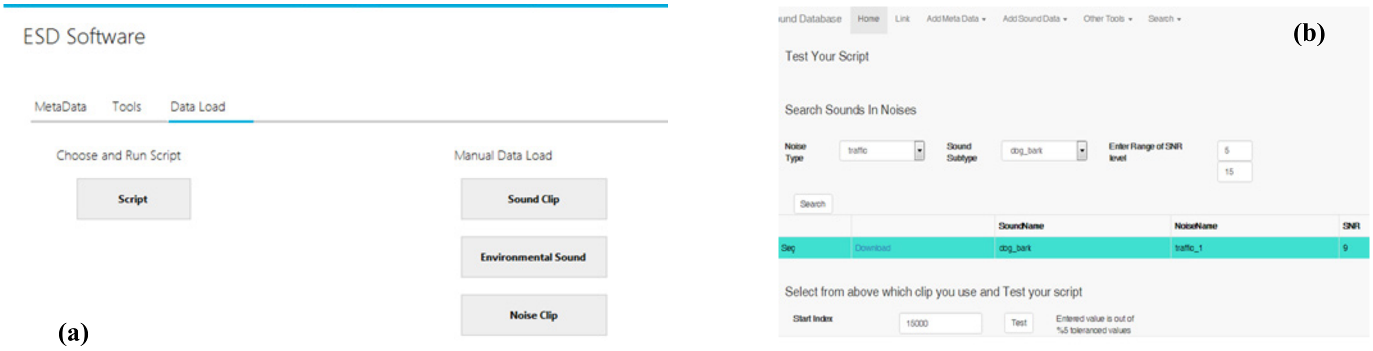
Although our goal of designing this database is for ESR, other areas that deal with environmental sounds can use it.

By improving the database by adding data from general databases mostly used and by adding more functionality we aim it to be a common database for research activities of ESR.

Conflict of Interest

The authors declare no conflict of interest.





		SoundName	NoiseName	SNR
Seç	Download	dog_bark	traffic_1	9

Figure 12 (a) Desktop application GUI for administrators to data load (b) WEB GUI for end user tools

References

[1] Y.Arslan and H. Canbolat, “A sound database development for environmental sound recognition”, Signal Processing and Communications Applications Conference (SIU), 25th, 2017.

[2] T. Ahmed, M. Uppal and A. Muhammad, “Improving Efficiency and Realibility of Gunshot Detection Systems”, IEEE, ICASSP 2013.

[3] P. Thumwarin, T. Matsuura and K. Yakoompai, "Audio forensics from gunshot for firearm identification", Proc. IEEE 4th Joint International Conference on Information and Communication Technology Electronic and Electrical Engineering Tailand, pp. 1-4, 2014.

[4] Chu, S.;Narayanan, S.; Kuo, C.-C.J.;Mataric, M.J.:Where am I? Scene recognition for mobile robots using audio features, in 2006 IEEE Int.Conf. on Multimedia and Expo. IEEE, 885–888, 2006.

[5] Yamakawa, N.; Takahashi, T.; Kitahara, T.; Ogata, T.; Okuno,H.G.: Environmental sound recognition for robot audition using Matching-Pursuit, in Modern Approaches in Applied Intelligence, in K.G. Mehrotra, C.K. Mohan, J.C. Oh, P.K. Varshney & M. Ali (Eds), Springer Berlin Heidelberg, 1–10, 2011.

[6] Chen, J.; Kam, A.H., Zhang, J.; Liu, N.; Shue, L.: Bathroom activity monitoring based on sound, in Pervasive Computing, in H.W. Gellersen, R.Want, & A. Schmidt (Eds), Springer Berlin Heidelberg, 47–61, 2005.

[7] Vacher, M.; Portet, F.; Fleury, A.;Noury, N.: Challenges in the processing of audio channels for ambient assisted living, in 2010 12th IEEE Int. Conf. on e-Health Networking Applications and Services (Healthcom), IEEE, 330–337, 2010.

[8] Wang, J.-C.; Lee, H.-P.; Wang, J.-F.; Lin, C.-B.: Robust environmental sound recognition for home automation. Automation Science and Engineering, IEEE Transactions on, 5 (1) (2008), 25–31.

[9] Bardeli, R.;Wolff,D.; Kurth, F.; Koch,M.; Tauchert, K.-H.; Frommolt, K.-H.: Detecting bird sounds in a complex acoustic environment and application to bioacoustic monitoring. Pattern Recognit. Lett., 31 (12) (2010), 1524–1534.

[10] Weninger, F.; Schuller, B.; Audio recognition in the wild: static and dynamic classification on a real-world database of animal vocalizations. in 2011 IEEE Int.Conf. on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2011, 337–340.

[11] P. Foggia, A. Saggese, N. Strisciuglio, M. Vento, and N. Petkov. Car crashes detection by audio analysis in crowded roads. In Advanced Video and Signal Based Surveillance e (AVSS), 2015 12th IEEE International Conference on, pages 1-6, Aug 2015.

[12] J. Rouas, J. Louradour, and S. Ambellouis, "Audio Events Detection in Public Transport Vehicle," Proc. of the 9th International IEEE Conference on Intelligent Transportation Systems, 2006.

[13] R. Radhakrishnan and A. Divakaran, “Systematic acquisition of audio classes for elevator surveillance,” in Image and Video Communications and Processing 2005, vol. 5685 of Proceedings of SPIE, pp. 64–71, March 2005.

[14] P. K. Atrey, N. C. Maddage, and M. S. Kankanhalli, “Audio based event detection for multimedia surveillance,” in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’06), vol. 5, pp. 813–816, Toulouse, France, May 2006.

[15] Nakamura, S.; Hiyane, K.; Asano, F.; Nishiura, T.; Yamada, T.: Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition, in LREC, 2000.

[16] DCASE 2016 WEB site available at: http://www.cs.tut.fi/sgn/arg/dcase2016/ (27.09.2017 last accessed)

[17] DCASE 2017 WEB site available at: http://www.cs.tut.fi/sgn/arg/dcase2017/ (27.09.2017 last accessed)

[18] K. J. Piczak, "ESC: Dataset for environmental sound classification", Proceedings of the ACM International Conference on Multimedia, 2015.

[19] Justin Salamon , Christopher Jacoby , Juan Pablo Bello, A Dataset and Taxonomy for Urban Sound Research, Proceedings of the 22nd ACM international conference on Multimedia, November 03-07, 2014.

[20] Brian Gygi and Valeriy Shafiro, “Development of the Database for Environmental Sound Research and Aplication (DESRA):Design, Fuctionality and Retrieval Considerations”, EURASIP Journal on Audio, Speech, and Music Processing Volume 2010.

[21] http://www.auditorylab.org (28.09.2017 last accessed)

[22] W. W. Gaver, “What in the world do we hear? An ecological approach to auditory event perception,” Ecological Psychology, vol. 5, no. 1, pp. 1–29, 1993.

[23] A. L. Brown, J. Kang, and T. Gjestland. Towards standardization in soundscape preference assessment. Applied Acoustics, 72(6):387-392, 2011