



Noise signature identification using mobile phones for indoor localization

Sayde King¹ · Samann Pinder² · Daniel Fernandez-Lanvin³ · Cristian González García³ · Javier De Andrés⁴ · Miguel Labrador¹

Received: 3 June 2022 / Revised: 13 November 2023 / Accepted: 13 December 2023
© The Author(s) 2024, corrected publication 2024

Abstract

Indoor localization is still nowadays a challenge with room to improve. Even though there are many different approaches that have evidenced as effective, most of them require specific hardware or infrastructure deployed along the building that can be discarded in many potential scenarios. Others that do not require such on-site infrastructure, like inertial navigation-based systems, entail certain accuracy problems due to the accumulation of errors. However, this error-accumulation can be mitigated using beacons that support the recalibration of the system. The more frequently beacons are detected, the smaller will be the accumulated error. In this work, we evaluate the use of the noise signature of the rooms of a building to pinpoint the current location of a low-cost Android device. Despite this strategy is not a complete indoor localization system (two rooms could share the same signature), it allows us to generate beacons automatically. The noise recorded by the device is preprocessed performing audio filtering, audio frame segmentation, and feature extraction. We evaluated binary (determining if the ambient sound recording belonged to a specific room) and multi-class (identifying which room an ambient noise recording belonged to by comparing it amongst the remaining 18 rooms from the original 19 rooms sampled) classification methods. Our results indicate that the two Stacking techniques and K-Nearest Neighbor (KNN) machine learning classifier are the most successful methods in binary classification with an average accuracy of 99.19%, 99.08%, and 99.04%. In multi-class classification the average accuracy for KNN is 90.77%, and 90.52% and 90.15% for both Voting techniques.

Keywords Noise signature identification · Indoor localization · Audio processing · Acoustic signals · Feature extraction · Ambient sound

1 Introduction

Localization is the method of determining the position of an object in space. Particularly, indoor localization involves locating an object or person indoors through an indoor posi-

✉ Cristian González García
gonzalezcristian@uniovi.es

Extended author information available on the last page of the article

tioning system (IPS). The issue of localization is completely solved outdoors by GPS, yet it becomes a problem again once inside of a building [1]. Though the issue of indoor localization remains unresolved, its implementation varies such as through its usage within applications with position tracking and activity monitoring [2, 3], building management [4, 5], and it even extends to the Internet of Things (IoT) [6]. Although some of these approaches have evidenced as effective solutions, most of them are based on the use of supplementary hardware and infrastructure that increments implementation and maintenance cost, or require intrusive methods that involve user's collaboration and implication (like wearing specific devices, using face recognition, etc.).

An interesting alternative to those that require specific infrastructure are those based on inertial navigation. This approach does not require of previous calibration, nor installation of external infrastructure/additional hardware [7]. It uses the array of inertial sensors embedded in the smartphone (triaxial orthogonal accelerometer, gyroscope, magnetic field detector, barometric pressure sensor, etc.) to recognize turns, stationary times, walking or stairs (among others) that allow us to track the movement of a smartphone inside a building [3] and localize it with acceptable accuracy. Nevertheless, although these systems perform well in small buildings and short and simple trajectories, the accumulative error generated in each action recognition can mean a problem whenever the users perform long and complex tours (a visit to a museum, for example). This error can be corrected using beacons to re-calibrate the position. A common strategy is the use of easy to identify points of interest (like stairs, for example), but this approach is limited by the architecture of the building.

This led us to the need to increase the number of these beacons. The more beacons we have, the more frequently the accumulated error would be reset. In this context, the objective is not a complete IPS, but to be able to pinpoint specific positions inside a building. Several previous studies on indoor localization do so by means such as WiFi, Bluetooth, Ultrasound, Visible light, Radio Frequency Identification (RFID) [8], Acoustics, and Ultra Wide-band [6], each with their respective advantages and disadvantages.

Our approach is to make use of the ambient noise signature of the different spaces of a building to recognize where the user is. Thus, in this work, we evaluate the feasibility of using the noise captured with a regular low-cost smartphone device to pinpoint the room or corridor the user is in. This technique has been used with success in other research works [9–12], but with different objectives. This method requires no additional on-site infrastructure to perform noise signature recognition and holds potential to be applied in a supplemental manner towards achieving the goal of indoor localization. We recorded the ambient sound of several rooms of the Engineering Building II of the University of South Florida. After filtering the sample to isolate ambient noise, we trained the system using both binary classification -whether or not an audio sample belonged to a specific room- and multi-class classification, which room out of the 19 possible rooms, hallways, entries, and meeting spaces does the audio sample belong to. The results of the experiments we conducted yielded an accuracy of 77.84%, 90.77%, 86.73%, 68.2%, 83.66%, 10.42%, 89.02%, 89.94%, 87.85%, 88.17%, 90.52%, and 90.15% with the J48, KNN, MLP, Naive Bayes (NB), Support Vector Machine (SVM), AdaBoost, Random Forest, Bagging, two Stacking, and two Voting classification algorithms on the 19 by 19 room multi-class classification problem. Alternatively, with binary classification, an accuracy of 97.74%, 99.04%, 98.84%, 85.38%, 97.01%, 96.58%, 98.55%, 98.22%, 99.08%, 99.19%, 98.84%, and 98.8% were achieved for J48, KNN, MLP, NB, SVM, AdaBoost, Random Forest, Bagging, two Stacking, and two Voting classification algorithms, respectively.

The organization of this paper is structured as follows: Section 2 summarizes works related to feature extraction, machine learning classification and audio processing. Section 3

encompasses a detailed description of the system proposed in this research. Section 4 includes the experimental tests performed on the proposed system and the results. Section 5 serves as a conclusion of the results and a brief explanation of proposed future work. Section 6 discusses the limitations of this proposed work.

2 Related works

Recently, there have been studies that focus on indoor localization via acoustic signals but do so with approaches different from the approach employed in this paper [10–12]. For example, many studies have cited the research done by Azizyan et al. [13], which achieves localization via sound, light, WiFi, and color of locations. This is but one example of localization via acoustic signals performed by a method apart from the acoustic signals exclusively. Other studies implement indoor localization by analyzing the reverberation of a signal emitted by the device [14] [5]. Additionally, systems such as those designed by Sun et al. [1], and Jia et al. [5] implement systems and solutions to indoor localization that are not designed for use on a smartphone, making these systems inconvenient for the average person to use.

2.1 Classification systems using sound features

In his paper, Doğan [15] uses acoustic signal processing to classify different road conditions such as asphalt, gravel, stone, and snow-covered roads. He extracted features from the asphalt samples such as Power Spectrum (PSC) and Mel Frequency Cepstral Coefficients (MFCC), and classified them by using Linear Predictive Coding (LPC) and an SVM. He found that road conditions were classified with a 97.5% accuracy rate, and when artificial noise data (i.e., cars passing by and rain) was added to the sample, the success of classification for cars passing by and rain were 89% and 67%, respectively.

Likewise, Tradigo et al. [16] use voice acoustic features to indicate specific vocal diseases via mobile devices. Some of the extracted audio features include frequency (F0), jitter (J), Relative Average Perturbation (RAP), shimmer, Adaptive Noise Normalized Energy (ANNE), and Harmonic to Noise Ratio (HRN). They then trained various classifiers with both male and female voice data sets i.e., SVM, Naive Bayes Classifier (NBC), J48 and Multilayer Perceptron (MLP).

There are also several papers that used acoustic signal processing for detecting medical ailments of a patients' lungs and heart. For instance, Grønnesby et al.'s study [17] automates the detection of abnormal sounds such as crackles in lungs using a smartphone. They carried out their experiments with a 5-dimensional vector and extracted features including variance, range, sum of simple moving average (fine and coarse), and spectrum mean to classify the sound as normal or having crackles, achieving an accuracy of 86% with an SVM.

Other studies also used acoustic signals to evaluate sounds in different environments and on a variety of surfaces. For example, Zeng et al. [18] describe a method of determining whether a watermelon is ripe or unripe through the use of machine learning algorithms and mobile device microphones. They extracted features from their sound samples including sound-to-noise ratio (SNR), zero crossing rate (ZCR), short time energy (STE), Sub-band short-time energy ratio, MFCC, and Brightness, and trained an SVM for classification of ripeness, exceeding 89% accuracy.

Yang and Hsieh [19] trained a Recurrent Neural Network (RNN) and a Convolutional Neural Network (CNN), on acoustic signals of heart sounds preprocessed with Discrete

Fourier Transform (DTF), and the variance and std. deviation of DFT. Both networks yielded similar accuracy around 80%.

In their research, Lavner et al. [20] also use a complex CNN classifier and a logistic regression classifier, where it is shown that the CNN classifier has a considerable advantage over the logistic regression classifier in terms of results. One of the goals of this study was to construct a platform for conducting psychological research on co-regulatory patterns between a baby and its caregiver, with cry events being a primary variable as a predictor of attachment. This research is relevant to our goal because it proposes methods of detecting specific sound events, which may prove to be useful for us in determining key room features.

McLoughlin et al. [21] perform classification of sound events with Deep Neural Networks (DNN), SVM, Stabilized Auditory Images (SAI) and Spectrogram Image Features (SIF). They found that DNN performs the best when classifying robust events, yielding an accuracy of 92.58% on average for noise-corrupted samples and both SVM and DNN classifiers perform well in noise-free conditions.

In the study performed by Naronglerdrit et al. [4], the authors propose human activity recognition using mobile phone microphones and trained NB, KNN, C4.5 decision trees, SVM, and an MLP using WEKA. Naronglerdrit et al. also propose that clustering to represent features such as MFCC, spectral entropy, harmonics to noise ratio (HNR), and linear prediction coding coefficients to decompose audio wave forms. The results showed that the best results were achieved with an MLP model, yielding a 92.46% accuracy. This research study provides evidence for the usefulness of various classifiers for the task of audio classification given specific features.

SoundSense by Lu et al. was the first general purpose sound sensing system for resource limited phones [22]. This study classifies general sound types such as music and voice (coarse category classification) but also goes a step further to classify genres of music (intra-category classification) simply from an audio file. This work performs action recognition via ambient sound of the following activities: walking, driving, riding an elevator, and riding a bus. For audio preprocessing, Soundsense conducts frame admission control to remove frames with only white noise or silence by considering energy levels and spectral entropy. Low energy levels and high entropy would indicate white noise or silence. Features used for classification of coarse categories are: Zero Crossing Rate (ZCR), low energy frame rate, spectral flux (SF), spectral rolloff (SRF), spectral centroid (SC), bandwidth, normalized weighted phase deviation, relative spectral entropy, and variance. This study implemented a decision tree classifier with Markov-models for each activity. For intra-activity classification, MFCC are calculated and then converted into MFCC feature vectors to be used in a simple Bayes classifier to represent various ambient sound events or activities. This method of examining ambient sound for the purpose of activity recognition via classification is related to the goal of our study, as we use ambient sounds of various locations for classification. This study showcases an effective use of audio features in a classification task, and we ultimately extract many of the same features presented here in our approach. Although this issue shares commonalities to our goal and provides scalability for mobile devices, it does not make advancements towards indoor navigation.

Scarpiniti et al. [23] propose a novel real-time approach to classify small audio frames to identify work activity and watch remotely construction sites. To achieve this, they distributed acoustic sensors, create a Deep Belief Network (DBN), and obtained the features from MFCCs and applied six aggregate statistics on it. They obtain an overall accuracy up to 98%.

In [24], the authors use an occluding intra-aural device to obtain sounds and classify them in human nonverbal events (clicking of teeth/tongue, blinking, ...), which could be used

to monitor users' health. They used clustering (Gaussian Mixture Models (GMM) and K-means) and classification (SVM and Random Forest) methods. The features were obtained using MFCC and Auditory-inspired Amplitude Modulation (AAM), and Per-Channel Energy Normalization (PCEN), and concatenated their histogram level using difference combinations among them three. The GMM and SVM option, using MFCC and PCEN for feature extraction, obtained a performance of 81.5% in sensitivity and 83% in precision.

2.2 Classification systems using indoor localization

Hasegawa et al. [14] propose that the acoustic signals of nearby surface materials can be used to locate a smartphone. They extract MFCC using Fast Fourier Transform (FFT), and carried out the experiment in three different environments to test the classification effectiveness. They achieved a 89.2% accuracy when classifying six different types of phone placements using Random Forest (RF). This system differs from our proposed work since their system requires a short beeping sound to be produced from the phone that echoes off of its surroundings to determine the phone's location, whereas our goal is to simply record the sounds in a room to determine which room it is in.

Marron et al. [3] present a system for pedestrian tracking and activity recognition in indoor environments. The proposed system uses widely-available sensors that are commonly found in smartphones and similar devices, ensuring that no additional or external infrastructure is necessary. Ultimately, they achieve an overall accuracy of 91.06% in common human motion indoor placements. Their system is able to recognize actions such as walking, walking on steps, standing still, or using the elevator from a smartphone device. This study shows an improvement towards the simplification and ease of activity monitoring indoors; however, this study does not solve the issue of locating the user indoors or recognizing the space the user is in. Rather, the study focuses on the actions and activity of the user.

SoundLoc and SurroundSense systems are well-cited and foundational works for indoor localization. SoundLoc, a system proposed by Jia et al. [5], considers indoor localization by emitting a Maximum Length Sequence (MLS) and extracting kurtosis, direct to reverberant energy ratio, and spectral standard deviation as features and performing classification with a NBC. This system performed with an accuracy of 97.8%. This work differs from our approach since it classifies the "echo" of a signal the system generates itself rather than classifying rooms based on their ambient noise. Additionally, this approach was not implemented on a smartphone. SurroundSense, which was proposed by Azizyan et al. [13], executes the task of ambient fingerprinting via ambient sound, light, color and WiFi. The acoustic fingerprint was generated by extracting the signal amplitude. However, this signature was employed as a filter rather than directly used in classification. While this study provides critical contributions to this field, they perform localization with means other than sound alone.

In their paper, Du et al. [25] implement indoor localization Probabilistic Neural Networks (PNN), which requires less training than other machine learning techniques and yields results of high accuracy [1]. Du et al. do not perform indoor localization with ambient sound alone, but also use chat word sensing, WiFi, and the user's schedule uploaded from a server, as well as Acoustic Background Spectrum (ABS) and sparse MFCC (SMFCC) as features. This study does not completely accomplish our goal due to the use of other components needed to perform the localization. However, with PNN, their system yields a 70% accuracy.

Furthermore, The Acoustic Landmark Locator (ALL) proposed by Phillips et al. [2] extracts the frequency and power spectrum to develop an acoustic signature for a specific hall or corridor. Through training an Artificial Neural Network (ANN), they achieved an

accuracy for the ALL ranging between 71% and 99%. The lower classification accuracy was reportedly due to the presence of carpet in the rooms or voices in the audio segments. Our approach extracts and separates the foreground noises from the ambient sound to allow for a more holistic analysis of the ambient noise present in the room or corridor while reducing the error created from foreground sounds such as voices altogether.

Similarly, in their paper, Sun et al. [1] aim to solve the issue of Sound Source Localization indoors with a PNN with a Generalized Cross-Correlation Classification Algorithm (GCA) to extract sound features. This work compares other existing Sound Source Location (SSL) classification techniques to their own system for evaluation of performance, which was successfully increased through these techniques. Although this work does execute accurate classification of sounds, it solves a different issue regarding where a sound is emanating from within a room or space, rather than recognizing acoustic signatures.

The study conducted by Moore et al. [26] is fundamental with respect to the idea of ambient noise characterizing a particular room or space. The concept of a roomprint is presented, which is similar to a fingerprint and is invariant with time, positioning, and noise. This work focuses on geometric features, room acoustic parameters, and environmental sounds, as necessary components of a roomprint. Roomprints exploit unique features of a room. Moore et al. also distinguishes between two types of roomprints: a reference roomprint and a latent roomprint; a reference roomprint can include factors of the room that can be explicitly measured and a latent roomprint can be derived only from recordings of speech that are uncontrolled. Using environmental sounds as a component of a roomprint directly aligns with the goals of this paper, since we hope to use ambient sound for classification of a room against others. Moore et al. achieved an overall error rate of 32.6%; however, when presenting the results they obtained with the logarithm of frequency-dependent reverberation time as a feature for classification, an error rate of 3.9% was achieved. This improvement highlights the substantial difference selecting features that better represent the data can have on performance.

Molina et al.'s work [27] showcases an implementation of a fingerprint-based system for indoor localization, specifically in an airport environment. They focus primarily on radio-frequency (RF) based approaches including GPS, WiFi, Bluetooth Low Energy (BLE), and Radio-frequency Identification (RFID) for localization. In their approach, the authors propose a system which makes use of location fingerprinting to compare the Received Signal Strengths (RSS) from each wireless access point in an area with prerecorded values. Fingerprinting is performed via offline sampling and online location. In processing their data, they use KNN for data classification and matching. In analyzing the results of their approach to localization, the authors found that there is a benefit to using several RF based methods at once, in that the use of different technologies together is more accurate than WiFi alone. Although this study aims to perform optimized indoor localization, it differs from our research goal in that it strictly uses RF-based methods to perform localization and does not consider audio in determining indoor location.

In their study, Leonardo et al. [28] propose a framework that achieves indoor localization on a smartphone device alone, and focuses on the pervasive or ambient sound of a room or environment. Their proposed algorithm, SoundSignature, extracts acoustic fingerprints and performs classification with an SVM. They extracted features including the logarithm of each frequency, MFCC, and Spectral Features such as centroid, spread, skewness, kurtosis, slope, decrease, and roll-off. They then performed feature selection via the Sequential Forward Feature Selection algorithm [29] and employed an SVM for classification with a binary classification or one-versus-rest approach. Ten-fold cross validation was used and for data collection, sound samples were collected from 16 locations, and the recordings were split

into 5-second long non-overlapping windows. One data set was recorded on a day with the A/C on and the other with it off. Applying SoundSignature to the first data set with 10-fold validation yielded an accuracy of 90.28%, using the same validation on both data sets combined yields a 77.89% accuracy and the validation of one data set with the other yields an accuracy of 48.08%. This study has promising results and has implemented this system on a smartphone device. However, this study does not solve the issue of classification beyond binary classification.

Van Haute et al. [8] evaluate different solutions for indoor localization that are RF-based. Although this study was not geared towards audio-based solutions, important flags were raised by the conclusions of their experiments. Van Haute et al. call for standardized evaluation methods to provide a means to objectively compare different localization solutions in multiple conditions. Ultimately, since current systems aimed towards solving the issue of indoor localization vary so greatly, it creates room for error when attempting to apply these solutions on different environments. This was shown by implementing three different solutions on three different environments – the results varied greatly. This study displays the unintentional bias evident in current solutions – that the solutions are tailored to specific environments (e.g. offices with brick walls and carpet vs. industrial spaces with concrete and tile).

Song et al. [9] propose a framework to determinate the indoor area location without any other dedicated device and just using a smartphone. To achieve this, they had to build an environmental background audio of the rooms to extract their fingerprints using the Pearson Correlation and Long Short-Term Memory (LSTM), which is a type of RNNs. However, LSTM was the best one between a comparison with KNN, Back Propagation (BP), and Radial Basis Function Kernel (RBF). Besides, they have divided rooms in subareas to create a more precise localization. In this proposal, they have used 96 hours of uncompressed audio in 14 different rooms and in different times, recording using a smartphone too. They obtained a room accuracy localization of 97.64%. In our case, we have obtained an average of 99.19% just using almost 8 hours of audio files (25 minutes in each room) and using Stacking and binary classification.

2.3 Audio preprocessing methods

In their paper, Bayle et al. [30] introduce Karalk, a data set geared towards cover song identification (CSI) and singing voice analysis. Although this study is not focused towards indoor localization, it does consider the issue of audio processing and classification which are critical pieces of our larger issue – indoor localization.

This paper uses Karalk on the task of CSI with the Dynamic Time Warping Method. They used several frameworks and tools including MARSYAS¹, and Yet Another Audio Feature Extractor (YAAFE)² to extract audio features from each track. Features extracted are: chroma, MFCC, chords and keys, chroma and chord distances. The system achieves an accuracy of 84%-89% for three of the features. This paper performs sound analysis, feature extraction and classification between the different genders corresponding to voice, genres of song, and more. We also utilize YAAFE for feature extraction which is a toolbox for audio feature extraction [31].

The study by Fedele et al. [32] finds a solution for conducting structural health assessment of road pavements by studying the acoustic properties of the pavement and optimizing the

¹ <http://marsyas.info>

² <http://yaafe.sourceforge.net/index.html>

pavement managing process. This study uses MATLAB coding³ for acoustic signal analysis an audio preprocessing. [31]. The analysis was specifically targeted towards Power Spectral Density (PSD) for the purpose of categorizing uncracked, lightly cracked, or highly cracked slabs of pavement. PSD measures the distribution of a signal's power over its frequency. The study concluded that signatures of cracked pavement have a reduced frequency and increase in amplitude.

3 Approach

Our proposed approach is similar to the SoundSignature system explored by Leonardo et al. [28]. However, our method increases the number of features extracted and implements other classifiers that were more successful with this type of problem in other works, such as the J48 [16], KNN, MLP, NBC [5], and SVM [4]. Furthermore, we have added some ensemble methods like AdaBoost, Random Forest, Bagging, Stacking, and Voting. Our approach consists of the separation of the original audio samples, as well as feature extraction, but does not include feature selection.

3.1 Data collection and preprocessing

Data collection took place during afternoon hours across one week in one of the University of South Florida's Engineering buildings that mainly consists of labs, offices, a convenience market, and meeting spaces. We performed data collection in 19 locations within the building including hallways, entryways, labs, offices, conference rooms, the convenience market, and the lobby. Table 1 displays the category signifying the kind of space to which each of the 19 locations corresponds. Locations in the hallways, labs, entryways, and convenience market categories typically are spaces with tile flooring and brick walls. Locations under the offices or meeting spaces categories typically have carpeted flooring with brick walls. We did not restrict the passing or entry of individuals while collecting data to record locations in their natural state. These distinctions are important to this work when considering the way sound travels in these spaces.

The audio recordings of each location are 25 minutes long, which allowed us to obtain 300 five-second samples of each area. As shown in other related works, sample lengths often vary but the number of samples used has exceeded 100 [28] and have sometimes exceeded 100,000 [19]. In our case, this particular number of samples provides us with a total number of 5,681 samples which is a reasonable medium between a few hundred and hundreds of thousands of samples. A Google Pixel smartphone was used to record all samples with a simple audio recording Android Application, Easy Voice Recorder⁴. While recording, the smartphone was kept on a flat surface, often on wooden or metal surfaces. Recordings included sounds such as talking, footsteps, machines, air conditioning, and the sounds of doors opening and closing.

In order to clean the audio and remove extraneous sounds that may overshadow the ambient sound of the recorded location, we followed the same procedure applied by Fassbender and Jones to remove the interfering noise [33], and implemented by Adobe Audition auto heal feature. This method works by analyzing the audio spectrum around the selected noise. It then reduces the amplitude of the noise, not just by reducing the volume, but also by smoothing

³ <https://www.mathworks.com/products/matlab.html>

⁴ <http://www.digipom.com/portfolio-items/easy-voice-recorder/>

Table 1 Room Classifications

Offices	Research Labs	Entryways	Hallways	Meeting Spaces	Convenience Market
Office1	Lab1	Entry1	Hallway1	Conference1	Market
Office2	Lab2	Entry2	Hallway2	Conference2	
Office3	Lab3	Entry3	Hallway3	Conference3	
Office4	Lab4	Entry4			

the spectral shape around the edges⁵. This process likely involves some form of spectral subtraction, where the spectral content of the noise is estimated and then subtracted from the noisy signal. We applied the process to the entire audio file for each location in 12-second non-overlapping windows.

3.2 Feature extraction

Since YAAFE provides such an expansive array of possible features to be extracted from an audio input sample, we found that it is useful for both extracting basic and more complex audio features, as it was applied in other studies with similar scenarios and constraints [34, 35]. This framework allows users to extract multiple features at once with a feature extraction plan while taking less CPU time than other tools like MARSYAS [31].

We use YAAFE to extract the following features: perceptual sharpness, perceptual spread, temporal and spectral centroid, temporal and spectral spread, temporal and spectral skewness, temporal and spectral kurtosis, spectral slope, spectral variation, spectral decrease, [36]; ZCR, spectral rolloff [37]; MFCC [38], LPC [39], energy and spectral flatness 1, and spectral flux 2. Table 3 provides descriptions of the features extracted in our study. The complete description of all features and the equations of the rest of the features can be found available online in YAAFE source repository⁶. Most features supported by YAAFE are also described in great detail in the work of Peeters [36], and this paper was used to complete Table 3. Equations for energy, spectral flatness, and spectral flux are provided in (1), and (2) respectively. Energy is expressed as the root mean square of an audio frame, x of the i the frame. Spectral flatness is computed as the ration between the geometric mean in the numerator and the arithmetic mean in the denominator. Spectral flux expresses the squared difference of normalized magnitudes of spectra between adjacent frames [40]. YAAFE's feature extraction is a two-step process, beginning with a feature plan parser. YAAFE takes the audio file and a Python script detailing the features the user wants to extract from the given audio file. The feature plan parser parses the Python script by feature and then defines the sequence of computational steps needed to extract each feature. From this, a dataflow graph is created for input to the second step, the data flow engine. Here, the series of computational steps are linked and executed using C++ component libraries on loaded the data component in accordance with the dataflow graph created in the first step. The output of the dataflow engine is the desired extracted features from the given audio file.

Regarding the features extracted, many of them such as spectral features [28], MFCC [25], kurtosis [5], energy [41], LPC [15], and ZCR [22] were also extracted in several of

⁵ <https://community.adobe.com/t5/audition-discussions/auto-heal-vs-spot-healing-brush-tool/td-p/10574082>

⁶ <http://yaafe.sourceforge.net/features.html>

Table 2 KNN 19 X 19 Classification Confusion Matrix

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
a	277	0	3	6	0	0	0	0	1	1	0	1	1	2	0	2	4	0	0
b	1	265	7	1	0	0	0	1	3	5	3	0	0	3	0	6	4	0	0
c	0	9	227	8	0	0	0	1	6	3	6	3	2	13	0	13	7	1	0
d	13	0	7	254	0	0	0	0	4	4	1	2	0	2	0	9	3	0	0
e	0	0	0	0	290	1	3	5	0	0	0	0	0	0	0	0	0	0	0
f	1	0	0	1	0	290	0	1	3	1	0	0	1	1	0	0	0	0	0
g	0	0	0	0	1	1	285	8	0	0	1	0	1	0	2	0	0	0	0
h	1	0	0	0	0	1	2	287	2	0	1	3	1	1	0	0	0	0	0
i	0	1	2	5	0	0	0	0	267	0	3	3	0	14	0	0	2	1	1
j	0	4	1	3	0	0	0	0	0	270	0	0	0	1	0	9	11	0	0
k	0	0	0	0	0	0	0	0	0	0	281	6	9	3	0	0	0	0	0
l	1	8	1	0	0	0	0	0	7	1	10	245	18	8	0	0	0	0	0
m	1	3	2	1	0	1	0	0	0	0	10	6	271	3	0	1	0	0	0
n	2	4	10	4	0	0	0	0	7	1	7	1	10	248	0	5	0	0	0
o	0	0	0	0	0	0	0	0	0	0	0	0	0	0	299	0	0	0	0
p	1	11	9	8	0	0	0	0	2	10	0	0	0	6	0	240	12	0	0
q	1	3	3	7	0	0	0	0	3	2	0	0	1	1	0	1	277	0	0
r	0	0	2	3	0	0	0	0	1	1	0	2	0	0	0	1	0	289	0
s	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	298

Table 3 Descriptions of Features Extracted

Feature	Description
<i>Perceptual Sharpness</i>	computes the sharpness of Loudness coefficients
<i>Perceptual Spread</i>	computes the spread of Loudness coefficients
<i>Spectral Flux</i>	computes the flux of spectrum between adjacent frames
<i>Energy</i>	computes the energy as root-mean-square of an audio frame
<i>Spectral Flatness</i>	computes global spectral flatness using the ratio between the geometric and arithmetic mean
<i>Linear Predictor Coefficients</i>	computes the LPC of a signal frame via auto-correction and the Levinson-Durbin Algorithm
<i>Spectral Variation</i>	the normalized correlation of spectrum between consecutive frames
<i>Spectral Slope</i>	computed by linear regression of the spectral amplitude, represents amount of decreasing of the spectral amplitude
<i>Zero Crossing Rate (ZCR)</i>	number of time-domain zero-crossings within a frame where the sign is 1 for positive arguments and -1 for negative arguments
<i>Temporal Centroid</i>	the time averaged over the energy envelope
<i>Spectral Centroid</i>	the barycenter of the spectral power distribution frequencies
<i>Spectral/Temporal Kurtosis</i>	measure of the flatness of a distribution around its mean value
<i>Spectral Rolloff</i>	the frequency so that 95% of the signal energy is contained below this frequency
<i>Spectral Spread</i>	spread of the spectrum around its mean value
<i>Spectral/Temporal Skewness</i>	measure of the asymmetry of a distribution around its mean value
<i>Mel-frequency cepstral coefficient (MFCC)</i>	represents the shape of the spectrum with very few coefficients
<i>Spectral Decrease</i>	represents the amount of decreasing of spectral amplitude

Table 4 MLP 19 X 19 Classification Confusion Matrix

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
a	270	3	3	17	0	0	0	1	1	0	0	0	1	0	0	0	2	0	0
b	6	215	16	1	0	0	0	0	7	12	2	11	5	8	0	9	5	2	0
c	0	11	206	10	1	0	0	0	5	0	7	3	9	25	0	15	5	2	0
d	7	1	6	247	0	0	0	0	10	7	0	0	0	10	1	3	4	3	0
e	0	0	0	0	291	2	2	4	0	0	0	0	0	0	0	0	0	0	0
f	0	2	0	0	3	284	3	5	0	0	0	1	0	0	1	0	0	0	0
g	0	0	0	0	7	1	279	8	0	0	1	0	1	1	1	0	0	0	0
h	0	2	0	0	4	7	0	281	0	0	0	1	0	1	0	0	0	2	1
i	1	2	4	2	1	1	0	0	251	0	3	6	2	21	0	2	1	1	1
j	3	4	4	12	0	0	0	0	1	256	0	0	1	2	0	11	5	0	0
k	0	0	0	2	0	0	0	1	3	0	269	8	8	6	0	2	0	0	0
l	0	2	5	0	0	0	0	3	7	0	16	227	22	11	0	2	2	1	1
m	1	1	2	1	0	2	1	0	4	1	11	21	236	16	0	1	0	1	0
n	1	1	13	2	0	2	1	1	7	2	5	5	11	244	1	0	0	3	0
o	0	0	0	0	0	0	0	0	0	0	0	0	0	0	298	0	0	1	0
p	2	1	20	7	0	0	0	0	1	9	1	0	0	4	0	226	25	3	0
q	7	6	5	11	0	0	0	0	2	4	1	0	1	1	0	19	242	0	0
r	0	2	1	1	0	0	0	0	1	0	0	1	1	3	0	0	0	289	0
s	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	297

Table 5 Voting I (Average of Probabilities) 19 X 19 Classification Confusion Matrix

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
textbfa	283	1	2	7	0	0	0	1	0	1	0	0	0	1	0	1	1	0	0
b	0	264	7	0	0	0	0	1	3	9	1	0	0	6	0	1	5	2	0
c	0	11	236	5	0	0	0	1	5	2	1	1	2	21	0	5	7	2	0
d	10	1	1	257	0	0	0	0	4	8	0	1	0	10	0	3	4	0	0
e	0	0	0	0	288	0	7	4	0	0	0	0	0	0	0	0	0	0	0
f	0	0	1	0	0	290	0	3	0	2	0	1	0	0	2	0	0	0	0
g	0	0	1	0	4	2	281	8	0	0	0	0	1	0	2	0	0	0	0
h	0	0	0	0	0	0	3	294	0	0	0	1	0	0	1	0	0	0	0
i	1	1	1	0	0	0	0	0	267	0	2	4	0	19	0	0	1	2	1
j	1	8	3	9	0	0	0	0	0	264	0	0	0	0	0	5	9	0	0
k	0	2	1	0	0	0	0	2	2	0	277	3	9	2	0	0	0	1	0
l	0	5	4	0	0	1	0	2	5	0	11	250	7	12	0	0	0	2	0
m	0	1	2	1	0	2	0	0	0	0	9	15	254	13	0	1	0	1	0
n	1	1	9	2	0	1	0	0	11	0	5	4	11	253	0	0	0	1	0
o	0	0	0	0	0	0	0	1	0	0	0	0	0	0	298	0	0	0	0
p	2	3	12	9	0	0	0	0	0	7	1	0	0	6	0	240	19	0	0
q	1	4	3	11	0	0	0	0	2	2	0	0	0	2	0	3	269	2	0
r	0	0	2	1	0	0	0	1	2	0	0	0	0	2	0	0	0	291	0
s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	299

Table 6 Voting 2 (Majority Voting) 19 X 19 Classification Confusion Matrix

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
a	275	2	2	11	0	0	0	1	0	2	0	0	0	2	0	1	2	0	0
b	0	267	4	0	0	0	0	1	3	9	1	0	0	5	0	1	5	3	0
c	0	12	230	4	0	0	0	1	4	3	3	0	2	25	0	6	7	2	0
d	9	2	2	250	0	0	0	0	10	8	0	1	0	10	0	3	4	0	0
e	0	0	0	0	292	0	5	2	0	0	0	0	0	0	0	0	0	0	0
f	0	0	1	0	1	289	0	5	0	2	0	1	0	0	0	0	0	0	0
g	0	0	1	0	5	2	279	9	0	0	0	1	0	0	2	0	0	0	0
h	0	0	0	0	0	4	2	290	0	0	0	1	1	0	1	0	0	0	0
i	0	2	1	0	0	0	0	0	266	0	2	4	0	21	0	0	1	1	1
j	3	9	4	10	0	0	0	0	0	260	0	0	0	0	0	6	7	0	0
k	0	2	1	0	0	0	0	2	3	0	276	5	7	2	0	0	0	1	0
l	0	9	2	1	0	0	0	1	6	0	11	247	7	13	0	0	0	2	0
m	0	1	2	0	0	2	0	0	0	0	8	19	248	17	0	1	0	1	0
n	1	1	9	3	0	1	0	0	13	0	4	4	11	252	0	0	0	0	0
o	0	0	0	0	0	0	0	1	0	0	0	0	0	0	298	0	0	0	0
p	2	4	13	8	0	0	0	0	0	7	1	0	0	5	0	243	16	0	0
q	1	2	4	10	0	0	0	0	3	2	0	0	0	3	0	7	266	1	0
r	0	1	1	1	0	0	0	1	1	0	0	1	0	2	0	0	0	291	0
s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	298

the studies discussed in the Related Works section of this paper (Section II). However, since YAAFE is able to extract different features in addition to those mentioned in previous papers, we utilized YAAFE 0.64⁷ for feature extraction to provide more information to the classifier for each sample, in efforts to yield an increase in performance. By manipulating the block and step size of the features to be extracted, where the block size is the unit of measure for frame size and step size is the step between consecutive frames, each feature was calculated for every five-second, non-overlapping window in the 25-minute audio file. Ultimately, we have 33 features per sample, totaling to 187,473 features for all samples.

$$energy = \sqrt{\frac{\sum_{i=0}^{N-1} x(i)^2}{N}} \quad \text{and} \quad S_{flatness} = \frac{\exp(\frac{1}{N} \sum_k \log(a_k))}{\frac{1}{N} \sum_k a_k} \quad (1)$$

$$S_{flux} = \frac{\sum_k (a_k(t) - a_k(t-1))^2}{\sqrt{\sum_k a_k(t-1)^2} \sqrt{\sum_k a_k(t)^2}} \quad (2)$$

3.3 Classification and Testing

WEKA 3.8.6⁸ was used to implement the following classification algorithms: J48, KNN, MLP, NBC, AdaBoost (Ada), Random Forest (RF), Bagging (Bag), Stacking (Stack), and Voting (Vote). We wrote a Python script to train an SVM and perform the binary classification task. The script also accepts an Attribute Relation File Format (ARFF) for input to the classification algorithms. As ARFF is the file format that is traditionally the input format into WEKA, we wrote our script to accept the same format to avoid tedious file format conversions. We decided on the first four classification algorithms due to their success in the following related works [4, 5, 16], which all employ one or more of the same classifiers. Additionally, we have added some ensemble algorithms like AdaBoost, Random Forest, Bagging, two Stacking, and two Voting because they could improve the results.

The J48 classifier is a pruned C4.5 decision tree [42], and was implemented with WEKA's defaults: a pruning confidence of 0.25 and a minimum of two instances per leaf. The KNN [43] performs distance weighting and selects a k value based on cross-validation. In our case, $k=1$ which has performed to us better than $k=3$. The MLP uses back-propagation to classify instances and all nodes are sigmoid for non-numeric classes⁹. Naive Bayes [44] is a classifier that chooses numeric estimator precision values based on the analysis of the training data. No modifications or changes were made to the classifiers' options NaiveBayes. Support Vector Machines find a hyperplane or a set of hyperplanes as a means to map the data presented. The hyperplanes fall between the different classes, making this algorithm effective for classification purposes. As SVM classifier has been used John Platt's Sequential Minimal Optimization (SMO) [45] with the default configuration, which replaces all missing values, transforms in binary the nominal attributes, and normalizes all of them.

AdaBoost [46] allows boosting classifier for nominal attributes. We used AdaBoostM1 with default configuration and DecisionStump as a Classifier. Random Forest combines trees depending on the values of a random vector [47]. Our Random Forest has the default configuration. Bagging generates different versions of one predictor to obtain an aggregated

⁷ <http://yaafe.sourceforge.net>

⁸ <https://www.cs.waikato.ac.nz/ml/weka/>

⁹ <http://weka.sourceforge.net/doc.dev/weka/classifiers/Classifier.html>

Table 7 Room Labels for Classification

Letter:	a	b	c	d	e	f	g	h	i	j
Room:	Conference1	Lab1	Office1	Office2	Entry1	Entry2	Entry3	Entry4	Conference2	Office3
Letter:	k	l	m	n	o	p	q	r	s	
Room:	Hallway1	Hallway2	Hallway3	Lab2	Market	Conference3	Office4	Lab3	Lab4	

Table 8 Accuracy for 19 X 19 Multi Classification

Algorithm	% Accuracy
J48	77.84
KNN	90.77
MLP	86.73
NBC	68.20
SVM	83.66
AdaBoost	10.42
Random Forest	89.02
Bagging	89.94
Stacking 1	87.85
Stacking 2	88.17
Voting 1	90.52
Voting 2	90.15

predictor [48], which we just configure to use RepTree as a classifier. Stacking allows the combination of several classifiers to do classification deducing the bias and the error rate [49]. In this paper, we have implemented two Stacking algorithms. In the first one, we just combine the algorithms used in [4, 5, 16]. In the second one, we added AdaBoost, Random Forest, and Bagging. In both cases, we use RepTree as a metaclassifier. Voting allows the combination of different classifiers to reduce the error based on the combination rule [50]. In this case, we have implemented two voting algorithms, both use all the algorithms, but the first one uses "Average of Probabilities" as a combination rule and the other one "Majority Voting".

Each test performed used a total of 5,681 instances. We performed multi-class classification (19-versus-19) and binary classification (one-versus-rest). The binary and multi-class classification tasks were implemented using all classification algorithms previously described in this subsection.

4 Results

In the multi-class testing, KNN performed the best with an accuracy of 90.77%. On the other hand, in the binary classification with an average accuracy of 99.19% the best algorithm is Stacking using J48, KNN, MLP, NB, SVM, AdaBoostM1, RandomForest, and Bagging as a classifiers. Table 9 shows the results of the binary classification problem. We have achieved an accuracy of 97.74%, 99.04%, 98.84%, 85.38%, 97.01%, 96.58%, 98.55%, 98.22%, 99.08%, 99.19%, 98.84%, and 98.8% were achieved for J48, KNN, MLP, NB, SVM, AdaBoost, Random Forest, Bagging, two Stacking, and two Voting classification algorithms, respectively.

The results for the 19 versus 19 classification problem are presented in Table 8 and the resulting confusion matrices from the two top performing classifiers that were used in previous researches – KNN and MLP are displayed in Tables 2 and 4, respectively. Tables 5 and 6 show the 19 vs 19 confusion matrix for the two best ensemble methods added in this research, where both are close to KNN. In addition, KNN improves by more than 4% the rest of the algorithms used in previous researches, and all ensemble methods except AdaBoost exceed

Table 9 % Accuracy for Binary Classification

	J48	KNN	MLP	NBC	SVM	Ada	RF	Bag	Stack1	Stack2	Vote1	Vote2
Conference1	98.45	99.31	99.22	92.42	98.2	98.09	98.97	98.69	98.28	99.35	99.15	99.14
Office1	97.35	98.58	98.14	70.27	94.72	94.73	97.96	97.77	98.83	98.98	98.38	98.31
Conference3	95.63	97.95	97.66	77.52	95.45	95.33	97	96.65	98.12	98.26	97.53	97.42
Office2	96.56	98.4	98.35	75.05	95.01	95.04	97.93	97.48	98.66	98.66	98.22	98.15
Entry1	99.17	99.81	99.82	87.18	99.41	98.57	99.68	99.36	99.81	99.83	99.8	99.8
Entry2	98.87	99.77	99.57	95.9	98.79	97.91	99.4	98.99	99.72	99.74	99.48	99.47
Entry3	98.77	99.66	99.79	91.79	98.73	97.2	99.16	99.03	99.77	99.75	99.33	99.32
Entry4	98.83	99.54	99.36	88.84	96.98	98.24	99.19	99.04	99.54	99.59	99.33	99.3
Lab1	97.33	98.74	98.81	83.42	96.56	96.45	98.08	97.7	99.04	98.99	98.57	98.5
Office3	97.89	98.86	98.65	84.83	98.23	97.2	98.79	98.42	98.86	98.91	98.96	98.92
Hallway1	97.31	99.01	99.02	87.66	97.92	94.84	98.46	97.94	99.3	99.31	98.94	98.9
Hallway2	96.58	98.56	97.88	78.27	94.72	94.74	97.71	97.51	98.57	98.61	97.98	97.95
Hallway3	96.61	98.62	98.1	79.31	94.74	94.74	97.5	97.17	98.72	98.88	98.04	97.97
Lab2	95.75	98.28	97.49	81.42	94.83	94.74	97.06	96.61	98.25	98.4	97.62	97.54
Market	99.68	99.96	99.97	99.73	99.86	99.75	99.94	99.65	99.96	99.94	99.96	99.96
Conference2	96.51	98.2	98.05	76.58	94.74	94.82	97.81	97.45	98.53	98.66	98.22	98.2
Office4	97.26	98.81	98.34	79.06	94.71	95.21	98.4	97.88	98.88	98.97	98.83	98.73
Lab3	98.69	99.77	99.74	93.17	99.53	97.92	99.44	99.10	99.79	99.79	99.63	99.62
Lab4	99.74	99.97	99.97	99.74	99.99	99.54	99.91	99.74	99.97	99.96	99.98	99.98
Average	97.74	99.04	98.84	85.38	97.01	96.58	98.55	98.22	99.08	99.19	98.84	98.8

the other algorithms except KNN. Table 7 provides the room corresponding to the labels on the confusion matrices.

In contrast, in binary classification, the best algorithms are Stacking 2, Stacking 1, and KNN, exceeding the 99% of accuracy. These results are displayed in Table 9, where it can be seen that SVM, like the other classifiers, outperforms the NB in this task across all performance measures. Although Table 9 indicates that certain rooms were more accurately classified with Stacking methods, this could be due to these methods have improved and reduced the bias and errors, after combining several classifiers including other ensemble methods that were not used in the other researches.

However, we can see that some rooms have very good accuracy with all the classifiers, including with NB, like the Market, and Lab4. It could be due to a lower amount of noise interference with the ambient sound in that particular room, hallway, or corridor. We found that rooms, hallways, or corridors with recordings of people talking or very distinct ambient sounds were classified with greater ease, as opposed to a recording in a room with a much quieter setting, which was more likely to be confused with another room, hallway, or corridor with similar conditions.

5 Conclusions and future work

The results shown above evidence the feasibility of our proposal. That is, the possibility of identifying the noise signature of an indoor space using only a low-cost device, without requiring any extra on-site hardware or infrastructure. Although in this work we used offline tools for the signal processing and model training, both the libraries we used and the models we generated in the training process could be added to a downloadable app. Even whether the performance of this architectural alternative would not reach the minimum levels of efficiency (running classification models in an Android device may require too many resources), there is room to adapt the model to a client-server architecture that could delegate the classification to the server-side, still avoiding any on-site infrastructure. Ultimately, this study has outperformed the work of Leonardo et al. [28] in the binary classification task with the SVM classifier with a difference of almost 7% accuracy. If we compare this study with Song et al. [9], they obtained an accuracy of 97.64% using LSTM-RNN while we obtained 99.19%

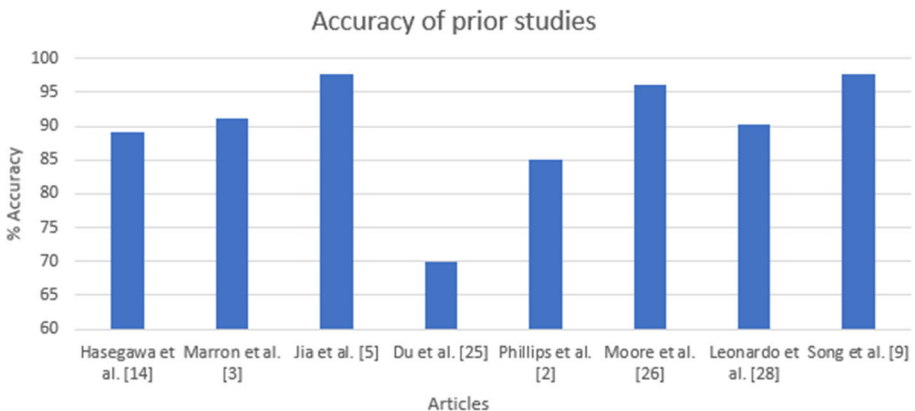


Fig. 1 Accuracy of prior studies

using a Stacking method and improved their result with our implementation of J48, KNN, MLP, Random Forest, Bagging, the two Stacking models, and the two Voting models. Furthermore, and as shown in Fig. 1¹⁰, our results also outperform in terms of accuracy those of the other prior studies that were reviewed in Section 2.2 Our study could be expanded upon to increase its impact and usefulness to the average person. Yet, as we discussed, it is not a complete indoor positioning system (two rooms sharing the same signature, for example), it can be used to automatically generate beacons to complement other alternatives, like the one proposed by Marrón et al. [3]. We believe that by extending our work, there could be potential applications for the system in indoor spaces like hospitals and malls. Future development of this work would include an increment of the variety of spaces and areas considered.

6 Limitations

This proposed study also has limitations that affect the feasibility of our solution. Firstly, in order for this system to be successful in any building, it would require the process outlined in Section 3 to be performed in every considered building. Unless this is performed by the building management, it places an unrealistic responsibility on the user in order to reap the benefits and convenience of this system. Secondly, since this system has not been tested on a smartphone, it is possible that it may require too much power to be carried out in real-time on a smartphone or other handheld smart device, forcing us to move to the client-server model alternative described above in the conclusions section. Lastly, the types of rooms and spaces we included in this experiment are not all-encompassing. Our study was limited to a single building, where despite the differences between each space, ultimately, the material used for the walls, flooring, and the purposes of the spaces were rather similar. It is possible that this system would not achieve the same performance when operating in an environment that is constructed and used differently. For example, a concrete, industrial setting that is mainly used for operating machinery may not be a suitable setting for our system. Furthermore, we also recognize that two or more rooms can share the same noise signature, such as offices and classrooms, and this is a problem that will have to be accounted for in future works with this system. Overcoming the limitations¹¹ addressed in this section would require sampling a larger number of buildings with different purposes and also a simplified process for preprocessing.

Acknowledgements The authors would also like to thank Raul Estrada and Jennifer Adorno for their insights and comments to improve the presentation of this paper and the quality of the study.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. This work was partially funded by the Department of Science, Innovation and Universities (Spain) under the National Program for Research, Development and Innovation (Project RTI2018-099235-B-I00). It was conducted for the Research Experience for Undergraduates program in the Computer Science Department at the University of South Florida.

Data Availability Statement The data is in ARFF files which are used in Weka to train and test the models. This data includes the rooms separated to do the binary classifications and a multiclass file with all the information. These ARFF files [51] can be found online in Mendeley Data.

¹⁰ In the case of Philips et al., we show the average of all their tests

¹¹ <https://data.mendeley.com/datasets/fm7cg3z3fj/1>

Declarations

Competing interest All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

Conflicts of interest The authors declare that they do not have any conflict of interest related to this journal.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References




1. Sun Y, Chen J, Yuen C, Rahardja S (2018) Indoor sound source localization with probabilistic neural network. *IEEE Trans Industr Electron* 65(8):6403–6413
2. Phillips L, Porter CB, Kottege N, D'Souza M, Ros M (2015) Machine learning based acoustic sensing for indoor room localisation using mobile phones. In: *Sensing technology (ICST), 2015 9th international conference on*, IEEE pp 456–460
3. Marron JJ, Labrador MA, Menéndez Valle A, Fernández Lanvin D, Rodríguez G, Martín, B (2016) Multi sensor system for pedestrian tracking and activity recognition in indoor environments. *International Journal of Ad Hoc and Ubiquitous Computing* 23 (1/2)
4. Naronglerdrit P, Mporas I, Sotudeh R (2017) Monitoring of indoors human activities using mobile phone audio recordings. In: *Signal processing & its applications (CSPA), 2017 IEEE 13th international Colloquium on*, IEEE pp 23–28
5. Jia R, Jin M, Chen Z, Spanos CJ (2015) Soundloc: Accurate room-level indoor localization using acoustic signatures. *Automation science and engineering (CASE), IEEE international conference on*, IEEE pp 186–193
6. Zafari F, Gkelias A, Leung K (2017) A survey of indoor localization systems and technologies. arXiv preprint [arXiv:1709.01015](https://arxiv.org/abs/1709.01015)
7. Harle R (2013) A survey of indoor inertial positioning systems for pedestrians. *IEEE Communications Surveys & Tutorials* 15(3):1281–1293. <https://doi.org/10.1109/SURV.2012.121912.00075>
8. Van Haute T, De Poorter E, Moerman I, Lemic F, Handziski V, Wolisz A, Wirstrom N, Voigt T (2016) Comparability of rf-based indoor localisation solutions in heterogeneous environments: an experimental study. *Int J Ad Hoc Ubiquitous Comput* 23(1–2):92–114
9. Song X, Wang M, Qiu H, Li K, Ang C (2019) Auditory scene analysis-based feature extraction for indoor subarea localization using smartphones. *IEEE Sens J* 19(15):6309–6316. <https://doi.org/10.1109/JSEN.2019.2892443>
10. Moghtadaiee V, Ghorashi SA, Ghavami M (2019) New reconstructed database for cost reduction in indoor fingerprinting localization. *IEEE Access*. 7:104462–104477
11. Ogiso S, Mizutani K, Wakatsuki N, Ebihara T (2019) Robust indoor localization in a reverberant environment using microphone pairs and asynchronous acoustic beacons. *IEEE Access* 7:123116–123127
12. Chen P, Liu F, Gao S, Li P, Yang X, Niu Q (2019) Smartphone-based indoor fingerprinting localization using channel state information. *IEEE Access* 7:180609–180619
13. Azizyan M, Constandache I, Roy Choudhury R (2009) Surroundsense: mobile phone localization via ambient fingerprinting. In: *Proceedings of the 15th annual international conference on mobile computing and networking*, ACM pp 261–272
14. Hasegawa T, Hirahashi S, Koshino M (2016) 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*, ACM pp 246–253
15. Doğan D (2017) Road-types classification using audio signal processing and svm method. In: *Signal processing and communications applications conference (SIU), 2017 25th, IEEE pp* 1–4

16. Tradigo G, Calabrese B, Macrí M, Vocaturo E, Lombardo N, Veltri P (2015) Voice signal features analysis and classification: looking for new diseases related parameters. In: Proceedings of the 6th ACM conference on bioinformatics, computational biology and health informatics, ACM pp 589–596
17. Grønnesby M, Solis JCA, Holsbø EJ, Melbye H, Bongo LA (2017) Machine learning based crackle detection in lung sounds. CoRR. [arXiv:1706.00005](https://arxiv.org/abs/1706.00005)
18. Zeng W, Huang X, Arisona SM, McLoughlin IV (2014) Classifying watermelon ripeness by analysing acoustic signals using mobile devices. *Pers Ubiquit Comput* 18(7):1753–1762
19. Yang T-cI, Hsieh H (2016) Classification of acoustic physiological signals based on deep learning neural networks with augmented features. In: Computing in cardiology conference (CinC), 2016 IEEE pp 569–572
20. Lavner Y, Cohen R, Ruinskiy D, IJzerman H (2016) Baby cry detection in domestic environment using deep learning. Science of electrical engineering (ICSEE). IEEE international conference on the, IEEE, pp 1–5
21. McLoughlin I, Zhang H, Xie Z, Song Y, Xiao W (2015) Robust sound event classification using deep neural networks. *IEEE/ACM Trans Audio Speech Language Process* 23(3):540–552
22. Lu H, Pan W, Lane ND, Choudhury T, Campbell AT (2009) Soundsense: scalable sound sensing for people-centric applications on mobile phones. In: Proceedings of the 7th international conference on mobile systems, applications, and services, ACM pp 165–178
23. Scarpiniti M, Colasante F, Di Tanna S, Ciancia M, Lee YC, Uncini A (2021) Deep Belief Network based audio classification for construction sites monitoring. *Expert Syst Appl* 177(March):114839. <https://doi.org/10.1016/j.eswa.2021.114839>
24. habot P, Bouserhal RE, Cardinal P, Voix J (2021) Detection and classification of human-produced non-verbal audio events. *Appl Acoust* 171:107643. <https://doi.org/10.1016/j.apacoust.2020.107643>
25. Du J, Chen W, Liu Y, Gu Y, Liu H (2013) Catch you as i can: indoor localization via ambient sound signature and human behavior. *Int J Distrib Sens Netw* 9(11):434301
26. Moore AH, Brookes M, Naylor PA (2013) Roomprints for forensic audio applications. Applications of signal processing to audio and acoustics (WASPAA). IEEE Workshop On, IEEE pp, pp 1–4
27. Molina B, Olivares E, Palau CE, Esteve M (2018) A multimodal fingerprint-based indoor positioning system for airports. *IEEE Access* 6:10092–10106
28. Leonardo R, Barandas M, Gamboa H (2018) A framework for infrastructure-free indoor localization based on pervasive sound analysis. *IEEE Sens J* 18(10):4136–4144
29. Jain A, Zongker D (1997) Feature selection: Evaluation, application, and small sample performance. *IEEE Trans Pattern Anal Mach Intell* 19(2):153–158
30. Bayle Y, Marsik L, Rusek M, Robine M, Hanna P, Slaninova K, Martinovic J, Pokorny J (2017) Karalk: A karaoke dataset for cover song identification and singing voice analysis. 2017 IEEE International symposium on multimedia (ISM)
31. Mathieu B, Essid S, Fillon T, Prado J, Richard G (2010) Yaafe, an easy to use and efficient audio feature extraction software. In: ISMIR, pp 441–446
32. Fedele R, Praticó F, Carotenuto R, Della Corte F (2017) Structural health monitoring of pavement assets through acoustic signature. In: BCRRRA 2017 (Tenth international conference on the bearing capacity of roads
33. Fassbender E, Jones CM (2014). In: Ma M, Jain LC, Anderson P (eds) The importance and creation of high-quality sounds in healthcare applications. Springer, Berlin, Heidelberg, pp 547–566
34. Bisot V, Serizel R, Essid S, Richard G (2017) Leveraging deep neural networks with nonnegative representations for improved environmental sound classification. In: 2017 IEEE 27th international workshop on machine learning for signal processing (MLSP), pp 1–6
35. Gubka R, Kuba M (2013) A comparison of audio features for elementary sound based audio classification. The International Conference On Digital Technologies 2013:14–17
36. Peeters G (2004) A large set of audio features for sound description (similarity and classification) in the cuidado project
37. Scheirer E, Slaney M (1997) Construction and evaluation of a robust multifeature speech/music discriminator. In: 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, vol 2, pp 1331–13342
38. Davis S, Mermelstein P (1980) Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans Acoust Speech Signal Process* 28(4):357–366
39. Makhoul J (1975) Linear prediction: A tutorial review. *Proc IEEE* 63(4):561–580
40. Giannakopoulos T, Pikrakis A (2014) Chapter 4 - audio features. In: Giannakopoulos T, Pikrakis A (eds) Introduction to Audio Analysis, pp 59–103. Academic Press, Oxford. <https://doi.org/10.1016/B978-0-08-099388-1.00004-2>

41. Elgendi M, Bobhate P, Jain S, Guo L, Kumar S, Rutledge J, Coe Y, Zemp R, Schuurmans D, Adatia I (2015) The unique heart sound signature of children with pulmonary artery hypertension. *Pulmonary circulation* 5(4):631–639
42. Salzberg SL (1994) C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Mach Learn* 16:235–240. <https://doi.org/10.1007/BF00993309>
43. Aha DW, Kibler D, Albert MK, Quinlan JR (1991) Instance-based learning algorithms. *Machine Learning* 1991 6:1 6:37–66. <https://doi.org/10.1007/BF00153759>
44. John GH, Langley P (1995) Estimating continuous distributions in bayesian classifiers, pp 338–345 . [arXiv:1302.4964](https://arxiv.org/abs/1302.4964)
45. Platt JC (1998) In: Schoelkopf B, Burges C, Smola A (eds) Fast training of svms using sequential minimal optimization. A. Smola. <https://www.researchgate.net/publication/242503764>
46. Freund Y, Schapire RE (1996) Experiments with a new boosting algorithm, pp 148–156 . <http://www.research.att.com/>
47. Breiman L (2001) Random forests. *Mach Learn* 45:5–32. <https://doi.org/10.1023/A:1010933404324/METRICS>
48. Breiman L (1996) Bagging predictors. 24:123–140
49. Wolpert DH (1992) Stacked generalization. *Neural Netw* 5:241–259. [https://doi.org/10.1016/S0893-6080\(05\)80023-1](https://doi.org/10.1016/S0893-6080(05)80023-1)
50. Kittler J, Hatef M, Duin RPW, Matas J (1998) On combining classifiers. *IEEE Trans Pattern Anal Mach Intell* 20:226–239. <https://doi.org/10.1109/34.667881>
51. King S, Pinder S, Lanvin DF, García CG, Suárez JDA, Labrador M (2023) Noise Signature Identification (Ambient Sounds in the University of South Florida, EBII) .<https://doi.org/10.17632/fm7cg3z3fj.1>. <https://data.mendeley.com/datasets/fm7cg3z3fj/1>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Sayde King¹  · Samann Pinder² · Daniel Fernandez-Lanvin³  · Cristian González García³  · Javier De Andrés⁴ · Miguel Labrador¹

Sayde King
saydeking@usf.edu

Samann Pinder
samann_pinder@sfu.ca

Daniel Fernandez-Lanvin
dfanvin@uniovi.es

Javier De Andrés
jdandres@uniovi.es

Miguel Labrador
mlabrador@usf.edu

¹ Department of Computer Science and Engineering, University of South Florida, 4202 E Fowler Ave, Tampa 33620, Florida, USA

² School of Interactive Arts and Technology, Simon Fraser University, 8888 University Dr, Burnaby BC V5A 1S6, Vancouver, Canada

³ Department of Computer Science, University of Oviedo, C/ San Francisco, 3, Oviedo 33007, Asturias, Spain

⁴ Department of Accounting, University of Oviedo, C/ San Francisco, 3, Oviedo 33007, Asturias, Spain