Article

SlideWin: Integrating Machine Learning with Human Knowledge for Auditory Scene Recognition with Limited Annotated Data

Theo Gueuret [1], Beiyu Lin [2,*], Lingwei Chen[3], Margaret McMullin [4], Ruben Alberts [4], Xiaowei Jia [5], Joel Snyder [4]

1 University of Lille; tgueuret@live.fr
2 University of Oklahoma; beiyu.lin@ou.edu
3 Wright State University; lingwei.chen@wright.edu
4 University of Nevada, Las Vegas; mcmulm1@unlv.nevada.edu
4 University of Nevada, Las Vegas; alberr2@unlv.nevada.edu
5 University of Pittsburgh; xiaowei@pitt.edu
4 University of Nevada Las Vegas; joel.snyder@unlv.edu
* Correspondence: beiyu.lin@ou.edu

Abstract: Sound is an important modality in the perception and understanding of the environment. With the development of digital technology, massive amounts of smart devices in use everywhere collect sound data. Auditory scene recognition is an important topic to understand and distinguish collected sounds via smart devices in the surrounding environment. The existing classical and state-of-the-art machine learning (ML) models can predict an auditory scene with an accuracy that varies between 50−73%, due to both limited annotated samples as well as limited knowledge of feature designs and extractions. We propose a novel yet simple Sliding Window pipeline (SlideWin), which utilizes domain experts to extract and select features that best describe auditory scenes, leverages windowing operations to increase samples for limited annotation problems, and improves the prediction accuracy by over 18% compared to the current best-performing models. SlideWin can detect real-life indoor and outdoor scenes with a 91.4% accuracy. The results will enhance practical applications of ML to analyze auditory scenes with limited annotated samples. It will further improve the recognition of environments that may potentially influence the safety of people, especially people with hearing aids and cochlear implant processors.

Keywords: Auditory Scene Recognition; Machine Learning; Small Sample Data

Citation: . . *Journal Not Specified* **2024**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

1. Introduction

In 2018, approximately 22 billion connected devices were in use around the world. In 2025, it may be around 38.6 billion [1]. As we can see the increasing use of smart devices, we anticipate an increasing amount of data collected from those smart and connected devices. An urgent question is how to effectively extract information from those data and leverage them to help people live a better life. Among various types of data collected by devices, the sound is an important modality to perceive and understand the environment. This work aims to design new algorithms to better understand and distinguish the surrounding environment by analyzing sounds collected via those devices, which is denoted as auditory scene recognition (ASR). In 2018, approximately 22 billion connected devices were in use around the world. In 2025, it may be around 38.6 billion [1]. As we can see the increasing use of smart devices, we anticipate an increasing amount of data collected from those smart and connected devices. An urgent question is how to effectively extract information from those data and leverage them to help people live a better life. Among various types of data collected by devices, sound is an important modality to perceive and understand the environment. This work aims to design new algorithms to better understand and distinguish the surrounding environment by analyzing sounds collected via those devices,

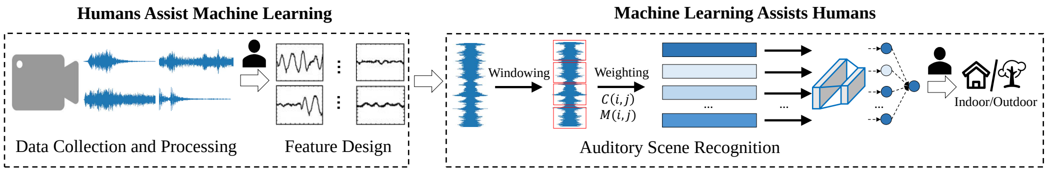

Figure 1. The overview of SlideWin for data-limited auditory scene recognition.

which is denoted as auditory scene recognition (ASR).

Auditory scene recognition has been applied to diverse areas, including context-aware services, intelligent wearable devices, robot sensing, and robot hearing [2–4]. It has also complemented research in other domains, such as multi-modal data fusion problems, detecting auditory events, and classifying audio from different people [5,6]. For example, ASR can improve the performance of sound-event detection, especially for hearing aids and cochlear implant processors, by providing a priori information about the probability of events [7].

Machine Learning has been extensively researched and commonly used in diverse disciplines. However, training advanced machine learning models, such as deep learning models, often require large amounts of data [8]. In many real-life applications, such as assisting people with hearing aids and cochlear implant processors, high-quality labeled samples are often difficult, or expensive to acquire [9], especially when we consider providing real-time personalized assistance with the need to quickly adapt to the environment that a user has been exposed to. This kind of situation requires completely different labeled data to facilitate meaningful classifications. Our study of auditory scene recognition starts with collecting auditory scene data as ground truth. The initial study in scene recognition requires highly-trained psychologists to travel to diverse sampling locations and deploy equipment in specific environments to record auditory scenes. This often incurs personnel and equipment costs for data that may not improve model performance. To make the data collection process efficient, such as capturing and enriching the properties and characteristics of numerous auditory scenes, we aim to build up a data-driven method that helps domain experts in determining the types and setting environment of audio data to be collected. Thus, the data collection process can be optimized by training ML models. We call this part **Machine Learning Assists Humans**.

Furthermore, in auditory scene recognition, the accuracy of the existing classical ML methods varies between $50-73\%$. These methods rely on different feature designs from machine learning experts but do not leverage the knowledge of domain experts. Alternatively, neural network models, such as Convolutional Neural Networks (CNNs) and Region-based Convolutional Neural Networks (R-CNNs), can automatically extract features from audio data and have been used in ASR, achieving the accuracy of 63% [10]. When applying ML methods to different areas, it is challenging to design and extract features that best describe the type of data and capture the properties of those data. Collaborating with domain experts to select and design those features offers the possibility of leveraging domain knowledge to improve machine learning models, with the goal of training an intelligent system to automatically learn and offer predictions later. In this study, we work with psychologists with expertise in auditory perception and cognition to design and extract features from auditory scenes. We call this part **Humans Assist Machine Learning**.

With machine learning and humans assisting each other, we denote it as **Integrating Machine Learning with Human Knowledge**. It can drastically increase data efficiency, improve the reliability and robustness of machine learning, and develop explainable intelligence systems. This offers the potential of instilling enormous amounts of human knowledge and the power of machine learning to achieve high performances, smooth human and ML interactions, and advance ML decisions to be understandable and interpretable to a larger and broader audience.

With this in mind, in this paper, we design a novel yet simple Sliding Window pipeline (SlideWin). It first utilizes domain knowledge to extract and select features that best describe auditory scenes. Then, it leverages windowing operations to generate samples from limited annotated audio data to better train a backbone classifier, such that the test prediction for each auditory sample can be derived by aggregating the outputs from its windows. Experimental evaluations demonstrate the effectiveness of SlideWin in recognizing indoor and outdoor scenes for real-life datasets. Our study includes three parts:

- **Machine Learning Assists Humans**. Make the limited auditory data more efficient by developing a data-driven method to assist domain experts.
- **Humans Assist Machine Learning**. Bridge domain expertise in auditory scene detection and machine learning to present categorized knowledge and its representations of auditory scene data.
- **SlideWin**. Design SlideWin pipeline to integrate domain knowledge and windowing operations into traditional machine learning classifiers to deliver superior performance with its interpretability to humans. The overview of SlideWin is illustrated in Figure 1.

2. Related Work

2.1. Auditory Scenes

Researchers have studied and designed different elements of auditory scenes, such as harshness, size, complexity, appeal, and spectral regions, to represent auditory and source properties that identify the perceptually auditory information of environmental sound recognition [11–13]. For example, researchers use spectral regions to identify the sets of sounds between 1200 and 2400 hertz (Hz) as one feature to capture the environmental sound [14]. Researchers have also investigated the similarity ratings of environmental sounds to determine the major perceptual dimensions of those auditory data [15].

Traditional statistical approaches (e.g., linear combinations) have been used to study correlation coefficients to find the relationship between auditory scenes and designed features [16]. For example, the coefficient of determination R^2 for indoor/outdoor auditory scenes and the designed auditory features based on linear regression demonstrate that a strong relationship exists between scenes and designed features [17]. In addition, other researchers have derived auditory grouping methods by measuring and summarizing statistics of natural stimulus statistical features. These statistics were used to disclose previously unrecognized grouping phenomena, and offer a framework for investigating grouping in natural signals [18].

2.2. Machine Learning

While those studies achieved excellent results, building up a model to predict auditory scenes is still challenging. Researchers have explored diverse machine learning methods to perform auditory scene detection. For example, given the data from TUT Urban Acoustic Scenes Mobile 2018 Development with 864 10-second segments for each acoustic scene, the average prediction accuracy, a baseline system of auditory scene detection using CNNs, ranges from 45.1% ($\pm$3.6) to 58.9 % ($\pm$0.8) for different devices [19]. One possible explanation for the low accuracy might be the only two acoustic features that are used for training: analysis frame 40 mid-side stereo and log mel-band energies. To automatically extract features from the auditory data, researchers have proposed diverse state-of-the-art deep learning methods (e.g., fusing) for auditory scene detection [10,20–23], however, the best classification accuracy in those studies was 72.8%.

Although state-of-the-art deep learning can automatically detect features, it's a black box, hard to interpret, and requires lots of training time and training samples. Our work collaborates with psychologists to leverage their expertise in auditory perception and cognition in feature designs and extraction of auditory scenes, which establishes an efficient learning method with limited annotated samples. Not only can our method significantly reduce the data requirement, but it can also build explainable intelligence systems.

3. Problem Definition

Our objective is to integrate domain experts' knowledge of acoustic characteristics into machine learning algorithms to improve the algorithm's performance in auditory scene classifications, given limited annotated samples. We are provided with a series of input features for each acoustic data based on 200 collected audio data (more details can be found in the sub-section Data Collection and Processing). The input features f_i for each acoustic data sample i are a D-dimensional vector $f_i = (f_i^1, f_i^2, ..., f_i^D)$. The feature vector includes spectral, statistical, discrete event, and sequence features (more details can be found in the sub-section Feature Design).

In the proposed method, we extract features that summarize the key characteristics of each acoustic data using a fixed-sized sliding window method (more details can be found in the Section Methodology). For each window j in each acoustic data i with a total number of K windows, we have input features $F_i = \{F_i^1, F_i^2, ..., F_i^j, ..., F_i^K\}$ and their labels $Y_i = \{y_i^j\}$, where F_i^j is a D-dimensional feature vector $F_i^j = (f_i^1, f_i^2, ..., f_i^D)$ (f_i^D is defined in the last paragraph). As described later in the methodology and result sections, we also divide the acoustic data into separate training and hold-out sets.

4. Methodology

Our designed Sliding Window pipeline (SlideWin) aims to investigate the properties and characteristics of auditory scenes, recognize these scenes when they occur in diverse and various settings for which auditory data is collected, and map the state of the environment and the corresponding sound sources in time to a specific scene label from a set of labels (i.e., indoor and outdoor). As depicted in Figure 1, our methodology SlideWin is composed of three steps as follows:

- *Data Collection and Processing*: select the audio data that comprise the current scene and define the corresponding context.
- *Feature Design*: design features that best describe and capture the properties and characteristics of those scenes.
- *Auditory Scene Recognition*: window and weight audio data to facilitate data-limited auditory scene classification model training, which learns the properties of scenes and produces meaningful prediction results.

4.1. Data Collection and Processing

We collect 200 naturalistic auditory scenes recorded one minute per sample with a Zoom Q8 camcorder mounted on a tripod. Each scene includes date, time, cardinal direction, temperature (°F), sounds observed, and any additional notes about the recording. To avoid noises and outlier information during audio collection, we manually extract 4-second clips to not only best describe and represent the scene settings, but also include sound sources that best describe real-life environments. All scenes were matched for root mean square (RMS) amplitude. In addition, to avoid abrupt onsets and offsets at each auditory scene, we impose a linear on-ramp from zero amplitude on the first 10 millisecond (msec) and a linear off-ramp to zero amplitude on the last 10 msec, respectively.

In this pilot study, we focus on the binary classification problem, whether an auditory data represents an indoor or outdoor environment by combining domain experts' knowledge into machine learning.

4.2. Feature Design

We evaluate the mid-level acoustic information for auditory scene identification and categorize those acoustic features into different five groups, including envelope-based features, statistical features, spectral features, discrete event features, and sequence features. To quantitatively understand mid-level audio information and its impacts on auditory scene identification, we design and extract 35 features, and present a description of 16 selected features as follows. A full list of the 35 feature description can be found in [24].

- *Moment CENTROID*. The centroid is the first moment of the spectrum and is a measure of the average distribution of energy related to the overall sound and tone of the auditory scene.
- *Moment KURT*. The kurtosis (KURT) is the fourth moment of the spectrum and measures the intensity of the distribution of energy related to the quality of the auditory scene.
- *RvA PauseAdjRMS*. This refers to the long-term or pause-corrected root mean square (RMS), and indicates the amount of silence present within each auditory scene.
- *RvA OverallRMS*. It refers to the overall RMS amplitude.
- *MPitch*. Mean pitch is the first of six correlogram-based pitch measures, which computes the mean rate of sound (pitch) by autocorrelating in 16 msec sliding windows.
- *SDPitch*. The third correlogram-based measure of pitch is the standard deviation of the pitch (which governs how high or low a tone sounds). This variable determines the pitch standard deviation in 16 msec sliding windows.
- *MAXPitch*. Maximum pitch is the fourth correlogram-based pitch measure, and calculates the highest rate of sound in a 16 msec sliding window.
- *MEANPSal*. Salience is a measure used in psychology that refers to how distinguishable features are compared to background noise. Mean pitch salience evaluates how distinguishable the average rate of sound is compared to background noise by autocorrelating in sliding 16 msec time windows.
- *AUTOCOR MaxPeak*. This refers to maximum peak in the autocorrelation.
- *AUTOCOR MeanPeak*. It computes mean peak in the autocorrelation.
- *AUTOCOR SDPeaks*. This is standard deviation of peaks in the autocorrelation.
- *x250Hz*, *x500Hz*, *x1000Hz*, *x2000Hz*, *x4000Hz*. These features are RMS energy in octave-wide frequency bands. Each of them also measures the distribution of energy across frequencies.

4.3. Auditory Scene Recognition

As the data selection and collection parts, the feature designs are already described in the above Subsections, here we focus on the choice of sub-sequence selection techniques to empower the discriminatory capacity of ASR in order to significantly advance prediction performances. We consider the window-sliding approach.

We restructure each streaming auditory data by dividing them into sliding windows, a set of ordered and overlapping sub-sequences. The windows are defined as below Definition 4.3. Each auditory wave/data sample consists of a sequence of n acoustic data points $< e_1, ..., e_n >$. Auditory windowing identifies a set of k windows in an auditory wave S. We denote each window in S as S_j, for $j = 1, .., k$, and the set of windows as $\mathcal{S} =< S_1, ..., S_k >$, where $\cup_{j=0}^{j=x} S_j \supseteq S$. The window size might vary and we define window sizes as $w = \{w_1, ..., w_k\}$. Thus, each S_j is an ordered sub-sequence of S. Window S_j can thus be represented by the sequence $< e_j, ..., e_{j+w_j} >$ for $j = 1, ..., k$.

4.3.1. Size Based Windowing

Using a sliding-window algorithm, we employ ASR based on the learned mapping $g : S_j \rightarrow E$, where E is a set of the environmental scene labels (e.g., indoor or outdoor) and we utilize the majority of the labels within a window as the scene label for this window. There are several different windowing approaches. Here, we use the size-based windowing approach to define window size, which ensures that the created windows contain an equal number of auditory data points from a collected auditory wave. This is generally referred to as *size-based sliding windows*. Using this approach, each window S_j, for $j = 1, ..., k$ has a set of windows $P =< S_1, ..., S_k >$ with a fixed number of auditory data points. That is, the window size vector $w = \{w_1, ..., w_k\}$ has $w_i = w_j$, for any $i, j = 1, ..., k$.

The features extracted from auditory data points in each window provides a context for making an informed mapping. Because the windows are ordered and non-empty, each window can be mapped to a scene label. This mapping is very effective, especially classifying scenes in real time from streaming auditory data. It not only offers a simple approach to learn the scene models during training, but also reduces the computational complexity of ASR. In addition, given a continuous auditory streaming data, this ASR technique can also be used for detecting a specific scene in an environment with background noises and irrelevant sounds.

A sliding-window approach requires a given window size and possible weighted features in each window. Window sizes is based on the appropriateness for the context and the type of scenes that will be recognized. In our study, we design the window size as 23,520 data points in each window and 14 windows for each acoustic data / wave. That is because, in order to estimate accurately our designed features in each window of acoustic data, a minimum of the two cycles of acoustic frequency is required. For example, amplitude envelope characteristics of speech that are important for identifying words, prosody, and accent are often on the time scale of 4-12 Hz, whereas the fundamental frequency of vocal fold vibrations important for identifying individual speakers are typically over 100 Hz. Thus, if the windows are too short to include at least two cycles of such frequencies (e.g., if we want to estimate frequencies down to 100 Hz, our window size should be no shorter than 20 ms, which is 2 cycles of 100 Hz. That is because, $\frac{1}{100}\text{Hz} \times 2 = .020\text{sec} = 20\text{msec}$; but for lower frequencies like 10 Hz, our window size should be no shorter than 200 ms, i.e., $\frac{1}{100}\text{Hz} \times 2 = 200$ msec).

4.3.2. Weighting Auditory Data Points Within a Window

Drawbacks of this windowing method exists, as every approach has its pros and cons. For example, if the time lag is large across different scene data points, the relationship between designed acoustic features in a window and a label in this window might exist, but very weak. As a result, treating all the auditory features with equal importance may result in loss of recognition effectiveness. That is, data points and features in collected auditory data may need to be weighted within a window given their relevance to the label, especially for ASR.

Furthermore, it is not uncommon to have multiple sound sources recorded in each acoustic data. However, auditory data from two different sources performed by the different environment scenes may be grouped into a single window, thereby introducing conflicting influences for the classification of majority labels in a window. To resolve this concern, we weight auditory data points and features within the window to describe the relationship between audio features. The weighting method offers computational advantages over other methods and can perform in real time as it does not need the knowledge of future auditory scenes to predict history auditory scenes.

Once an auditory window S_j is defined, the next step is to extract a feature vector within this window that contains relevant auditory scene information (content as described in the Subsection Feature Design). However, one of the problems associated with size-based windowing is the sparseness. That is, auditory scenes may be widely spread part in a window. It is possible for a sequence of discrete auditory data points collected and recorded from the environment via our collecting method and devices. This kind of gaps would impact the value of the features we designed and extracted from each window in an auditory data. In order to reduce the influences of such audio data points on deciding the scene label for the majority audio data points, a weighting factor is applied to each auditory data point in the window based on its relative time to the both first and last audio data point in the window. Let $t_{j-w}, ...t_j$ represent the time stamps of the audio data points in window S_j. For each audio data pint e_j^i in S_j, we compute the difference between the time stamp of e_j^i and the time stamp of the first and the last audio data point in the window, e_j^1 and e_j^w, respectively. We then choose the maximum value between these two distances. The contribution, or weight, of audio event e_j^i can be computed using an exponential function as show in Eq. (1):

$$C(i) = e^{-X \max(t_i - t_1, t_w - t_i)} \quad (1)$$

where the value of X determines the rate of decay of the influence.

Similarly, in situations when auditory feature corresponds to the transition between two scenes (or in other settings when multiple auditory scenes are performed by more than one sources in parallel), the auditory feature occurring in the window might not be related to the auditory data point relevant to the scene. For example, when auditory feature represent the transition from Indoor to Outdoor, all the initial auditory features in the window come from Indoor, whereas the later set of auditory data is from Outdoor. While defining a scene of a such situation, the chances for a wrong or uncertain conclusion about the majority scene data of the window are higher. This problem can be addressed by defining a weighting scheme based on mutual information (MI) between the auditory scenes.

The MI measure reduces the influence of auditory features within the window that do not typically occur within the same time frame. MI is in general used as a measure for the mutual dependence of two random variables. For ASR, each individual feature (described in the Section Feature Designs) is a random variable. The MI or dependence between two features is then defined as the chance of these two features occurring successively in an auditory data stream. If f_k^i and f_k^j are two features in a window k, then the MI between them, $MI(i,j)$, of an auditory data stream with k windows, is defined as

$$MI(i,j) = \frac{1}{K}\sum_{k=1}^{K-1} \sigma(f_k^i, f_k^j), \tag{2}$$

where

$$\sigma(f_i, f_j) = \begin{cases} 0 & f_k^i \neq f_k^j \\ 1 & f_k^i = f_k^j \end{cases} \tag{3}$$

If two features are related to each other or two features highly represents a same scene, then the MI between these two features will be high. Similarly, if the features are not related or their values are very different in a same scene, then the MI between them will be low. The MI matrix is typically computed offline using sample data, while audio scene labels are not necessary, from the set of auditory features that will be utilized for ASR.

4.3.3. Feature Selections

We aim to build up a model as simple as possible to broaden and include a larger audience while also offering superior performance. Given our initially designed 35 features with the domain experts based on mid-level acoustic information, it is very likely to develop a complex model as more features are included in a model. We learn the importance of those features by performing a feature selection and reduce the redundant features that may decrease the generalization competence of classification models. We use Information Gain to evaluate each feature and rank the importance of those features, and use Random Forest to select a group of acoustic features. Combining the selecting results of both Information Gain and Random Forest, we use 16 features (as listed in Table 1) as representatives for acoustic data in each window.

4.3.4. Training Classification Model

The windowing operation significantly augments the labeled samples for auditory scene recognition model training. To perform the classification of Indoor / Outdoor auditory scenes, we employ both traditional machine learning algorithms (e.g., Support Vector Machine and Random Forest) and the state-of-the-art methods (e.g., Neural Networks) as our backbone classifier devised in SlideWin: (1) traditional machine learning methods generally require less training efforts yet make decisions more understandable and interpretable to humans; (2) Neural Networks may further extract higher-level information and patterns from the input data that may not be learned from the traditional methods. As such, given the windowed and weighted acoustic features, we mainly focus on the following

Feature Names	
Moments CENTROID	Moments KURT
RvA PauseAdjRMS	RvA OverallRMS
MPitch	SDPitch
MAXPitch	MEANPSal
AUTOCOR MaxPeak	AUTOCOR MeanPeak
AUTOCOR SDPeaks	x250Hz
x500Hz	x1000Hz
x2000Hz	x4000Hz

Table 1. A list of the selected acoustic features. The description of each selected feature is detailed in Feature Design.

thirteen algorithms for auditory scene classification model training: Random Forest (RF), AdaBoost (Ada), Decision Tree (DT), K-nearest neighbors (kNN), Support Vector Machine (SVM), XGBoost (XGB), Neural Network (NN), Artificial Neural Network (ANN), Long Short-Term Memory Network (LSTM), Gated Recurrent Unit (GRU), Convolutional Neural Network (CNN), Recurrent Convolutional Neural Network (RCNN), and Convolutional Recurrent Neural Network(CRNN).

The next question is how to choose data on which the supervised learning algorithm will be tested. Because we want the learned model to generalize beyond the data it has already seen and correctly classify new data points that it has not previously seen, the model is usually trained on one set of data and tested on a separate, "holdout" set of data. In the context of ASR, the question is how to select subsets of available data for training and testing. Here we describe two techniques that are commonly applied to this process.

The first method is $k-$fold cross validation. this method is effective when the amount of labeled data is limited because it allows all of the data points to play a role both in training and testing the learned model. With this approach, the set of data points is split into k non-overlapping subsets. The model is trained and tested k times and the performance is average over the k iterations. On each iteration, one of the k partitions is held out for testing and the other $k-1$ partitions are used to train the model. The choice of k varies, common choices are 3-fold (this is particularly common when there are few data points available because each partition should ideally have at least 30 data points) and 10-fold cross validation. In this study, our performance evaluation is based on a 3-fold cross validation.

While cross validation is a popular validation technique for machine learning algorithms, its use is trickier when applied to sequential data (such as in our case of auditory scene data). This is because the long contiguous sequence must be separated into individual data points and subsets of data points. As a result, some of the contexts are lost that may be captured by some algorithms when training the model. When activities are pre-segmented, the segments can represent individual data points. When a sliding window approach is used, individual windows represent the data points. Some alternative selections are to separate the data sets by time boundaries such as each minute in order to retain much of the sequential relationships. This method also allows the ASR algorithm to be tested for its ability to generalize to new seconds. Leave-one-out testing can be used to train the data on contiguous sequential data and test it on held-out data from the end of the sequence. The length of the training sequence can iteratively increase so that eventually all of the available data is used for both training and testing.

While most methods choose a holdout set for testing either through random selection or at the end of a sequence, holdout selection can also be performed strategically to demonstrate the generalizability of the ASR algorithm over selected dimensions. For example, an algorithm can be trained over multiple auditory scene data, selecting one auditory scene data as the holdout set. Similarly, entire scene data can be held out to determine how the algorithm performs on a previously-unseen scene class.

4.3.5. Inference through Ensemble and Performance Metrics

After the windowing operation, each acoustic data sample is presented as a set of windows. Based on the trained auditory scene classification model, we can easily proceed with inference for test data by feeding the features of its windows and obtain a set of prediction outputs. Accordingly, we deploy a majority-voting ensemble to aggregate individual outputs and approximate the final prediction result. With the use of ensemble, the inference may provide some additional advantages beyond that provided by the trained model: (1) improving the classification performance, and (2) enhancing the inference resilience against noisy data. As for performance evaluation, we use the most common metric (i.e., accuracy) for our binary task to indicate the overall effectiveness of auditory scene classification. To guide multiple-epoch Neural Network training using gradient descent, we further use mean squared error (MSE) as the loss function $\mathcal{L}$ to evaluate training and validation performance, which is defined as follows:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{4}$$

where $\hat{y}_i$ and y_i denote the model prediction and the ground truth label respectively, and N is the training batch size.

5. Experiments and Results

As we have different types of data (raw, a full list of features, and windowed selected features), we use the thirteen ML algorithms to quantify and characterize differences in auditory scene classification. In particular, we performed the following experiments:

Experiment 1: Use the raw auditory data and implement the existing classical and state-of-the-art ML methods for auditory scene recognition of indoor and outdoor environments. Not only the results would demonstrate the performance of existing algorithms for this specific auditory dataset, but also provide a baseline for comparisons between the existing methods and our proposed method.

Experiment 2: Design 35 features of the auditory raw data with domain experts (without any sliding windows) and use those features for indoor and outdoor environment recognition via existing ML methods. The results offer a better understanding of the impacts of domain experts on feature designs.

Experiment 3: Given the raw auditory data, implement the SlideWin pipeline to recognize indoor and outdoor environments. This experiment helps us better understand the impacts of both SlideWin and feature designs by domain experts on environment recognition.

Experiment 4: Combine SlideWin (with 14 initial windows) and initially designed 35 features to identify indoor and outdoor auditory scenes. The results show that our proposed pipeline, integrating domain expertise and windowing operations into ML methods, delivers superior performance.

Experiment 5: Select the optimal 16 features from the initially designed 35 features and implement Experiment 4 on those selected features. This experiment shows that we can use ML methods to select features that greatest impact on auditory scene recognition while maintain the similar recognition accuracy of audiotry scenes.

Experiment 6: Vary the number of windows while meeting the requirements of minimum window size (discussed in Size Based Windowing) for both Experiments 4 and 5. This experiment demonstrates the stability and robustness of the proposed SlideWin regardless of the different numbers of windows or the optimized features, as long as the window size meets the minimum requirement.

5.0.1. Analyze Raw Acoustic Data

We directly implement both classical and state-of-the-art ML models to the raw auditory data (Experiment 1) shown in Fig. 2. As we can see in Fig. 2, the classification

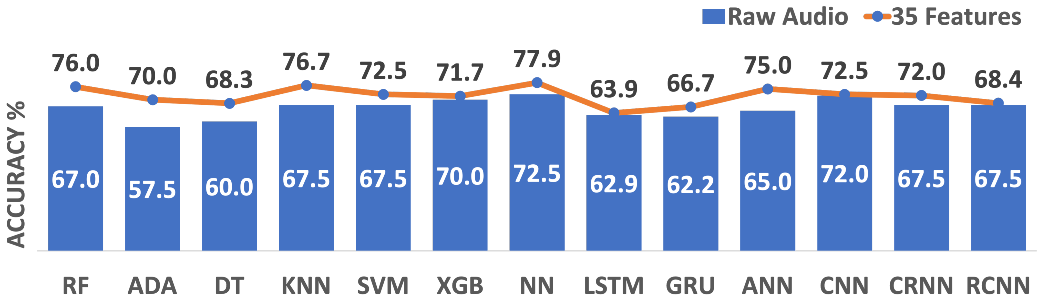

Figure 2. Train thirteen ML models on: raw data shown as blue bars (Experiment 1), and 35 features shown as the orange line (Experiment 2).

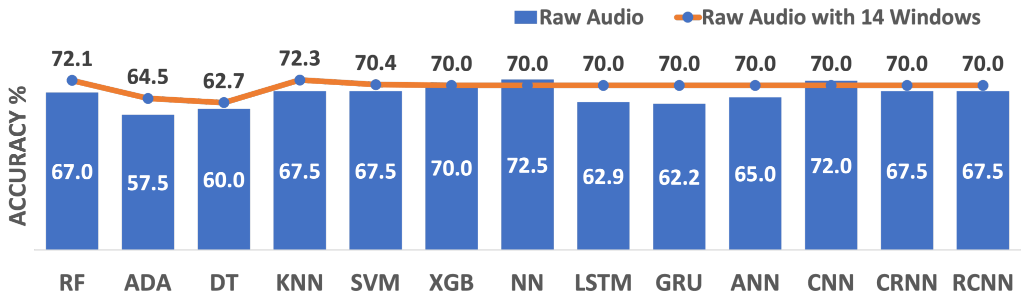

Figure 3. Trained thirteen ML models on: raw data (Experiment 1), and raw audio data with 14 windows (Experiment 3).

accuracy ranges from 57.5% to 75.2%, which is consistent with the results of auditory scene detection with different datasets [10] as discussed in the section Related Work. One possibility for the low accuracy to predict auditory classifications based on raw data is that those ML models have not extracted or learned the key properties from the raw auditory data. The other possibility is that the sample size, annotated 200 auditory data, may be not large enough to help ML algorithms learn and generalize. This leads to Experiment 2.

5.0.2. Analyze A Full List of Auditory Features

Working with domain experts offers us the opportunity to understand the key characteristics of auditory data. We hypothesize that sing those key characteristics as features of auditory data may help improve the predicting accuracy. We examine our hypothesis via Experiment 2 as shown in Fig. 2. The accuracy of Experiment 2 varies from 63.9% to 77.9%. Based on Fig. 2, we notice that models perform better by learning from the extracted 35 features instead of from the raw auditory data. For example, the RF algorithm improves the accuracy by 9% using the extracted 35 features than the raw data. Based on the extracted features, the prediction accuracy of NN is 77.9%, improves by 5% compared to the existing best-performing models (prediction accuracy is 72.5%).

While the improvement of the classification accuracy given the designed features is promising, extracting features from an entire auditory wave / data may over look or over smooth the variances and fluctuations of those features that represents scenes. This leads to our Experiment 3.

5.0.3. Analyze Selected Auditory Features in Each Window

In order to catch the detailed information of each auditory scene, we employ our SlideWin first to raw data for a baseline in Figure 3, and then to 35 features extracted from each window via 14 sliding windows as shown in Figure 4 (detailed numbers are in Table 2). In Figure 3, we notice that our SlidWin helps the majority (11 out of 13 models)

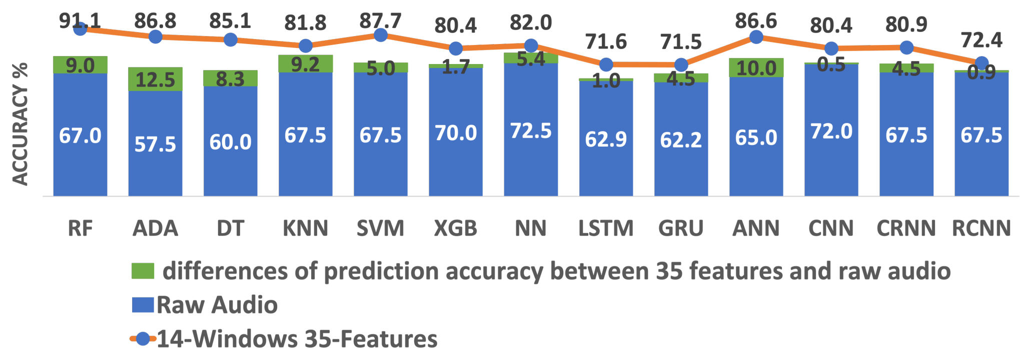

Figure 4. Trained thirteen ML models on: raw data (Experiment 1), and 14 initial windows with 35 designed features in each window (Experiment 4).

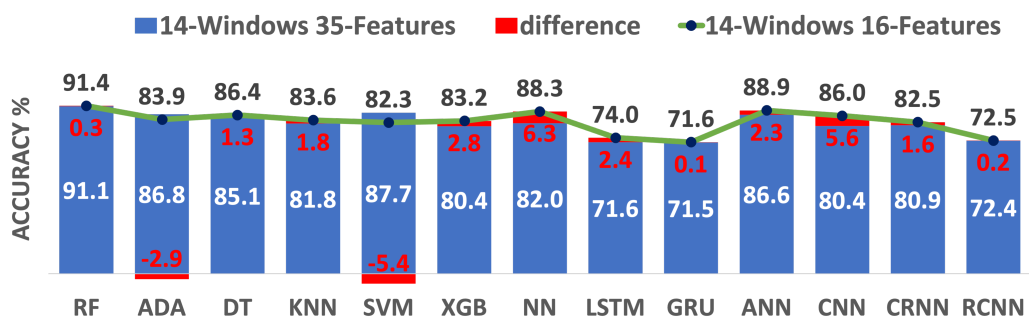

Figure 5. Trained thirteen ML models on: on raw data (Experiment 1), selected 16 features within 14 windows (Experiment 5).

of the selected ML models improve the performance, compared with those models on the raw audio data, and the improvements range from 0% to 7.8%. When we look at the performance of SlideWin on extracted features on each window in Figure 4, we notice that the performance based on SlideWin is way much better than those on raw data. The improved performance (in the orange line with blue dots) ranges from 4.85% to 29.28%.

Comparing the results between a total extracted 35 features from the raw data (adding the values of green and blue bars as shown in Experiment 2 in Figure 4) and each window with 35 features (orange line with blue dot Experiment 4 in Figure 4), we can see the performance improved ranging from 3.96% to 16.78%. The experiments demonstrate that our proposed SlideWin achieve superior performance than existing methods.

As we aim to the simplest model with best classification results to broaden and include a larger audience, more features may lead to more complex models that may not easy to interpret or understand. This leads to Experiment 5.

We perform a feature selection and reduce the redundant features to learn the importance of the features in auditory data. As shown in Figure 5, give a fixed number of windows, the performance differences between all 35 features in each sliding window and selected 16 features in each sliding window are in general (10 out of 13 models) similar (as the red bars in Figure 5) with ranging from -2.9% to 2.8%, though for SVM, NN and CNN, the difference is around 5% to 6% ish. Giving a fixed number of sliding windows, the selected 16 features in general achieve similar or slightly better performance than the entire 35 features.

Comparing the performance of our SlideWin to the current best-performing models (as described in the Section Related work), our SlideWin drastically deliver superior performance by over 18% and offers an accuracy of 91.4% for auditory scene classifications.

Table 2: The summary of the prediction accuracy (%) of thirteen ML models for different input feature settings.

Models	Feature Setting				
	Raw Data	35 Features	35 Features 14 windows	16 Features 14 windows	16 Features 20 windows
RF	67.00	76.00	91.14	91.35	91.04
Ada	57.50	70.00	86.78	83.92	82.72
DT	60.00	68.33	85.11	86.42	89.75
kNN	67.50	76.67	81.78	83.57	88.76
SVM	67.50	72.50	87.67	82.32	81.81
XGB	70 .00	71.67	80.36	83.21	86.70
NN	72.50	77.93	81.96	88.32	86.05
LSTM	62.87	63.88	71.63	73.96	72.66
GRU	62.19	66.66	71.49	71.63	72.64
ANN	64.99	75.00	86.60	88.89	89.96
CNN	72.00	72.50	80.39	86.00	85.02
CRNN	67.50	72.00	80.90	82.46	82.28
RCNN	67.50	68.39	72.35	72.51	72.01

5.0.4. Analyze Number of Windows and Window Size

To make sure that diverse window size and the number of window would not significantly impact the performance of our SlideWin, we perform the classification for selected features in each window and vary the window size from 14 to 99, while still guarantee that a minimum of the two cycles of acoustic frequency in each window. Our prediction results (as shown in Figure 6 of SlideWin given diverse window sizes. Given the mean accuracy for window sizes from 14 to 99 and the corresponding standard deviations (ranging from 0.5 to 3), we conclude that our SlideWin delivers superior performance of auditory scene detection, especially for Indoor / Outdoor scenes, and is stable and robust with different window sizes.

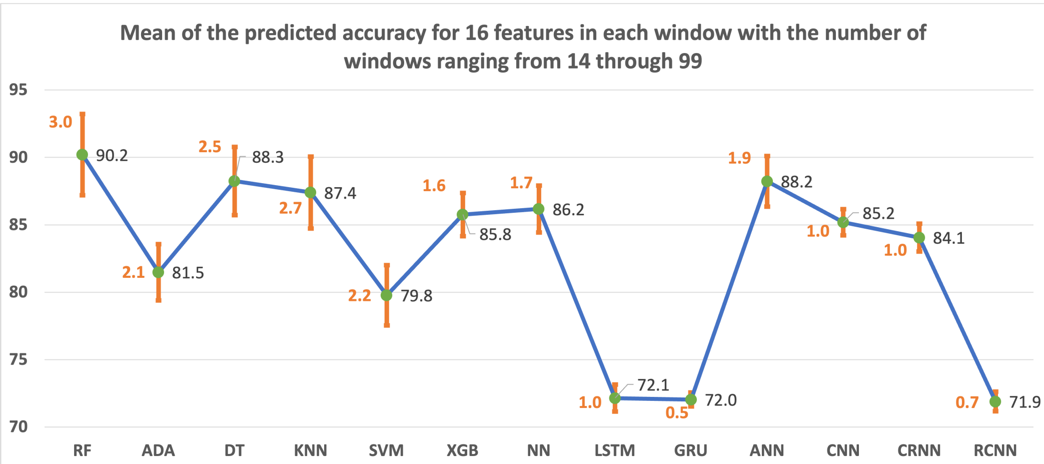

Figure 6. Mean of the predicted accuracy for the number of windows ranging from 14 through 99. There are 16 features in each window. The error bars represent the corresponding standard deviations, ranging from 0.5 to 3.0.

6. Discussion and Conclusions

As auditory scene recognition is an important topic in today's big data and pervasive environment, we propose a novel yet simple Sliding Window pipeline (SlideWin) for auditory scene recognition, that utilizes domain experts to extract and select features that best describe auditory scenes, leverages windowing operation to increase samples for limited annotation problems, and improves the prediction accuracy by over 18% compared to the current best-performing models.

SlideWin can detect real-life indoor and outdoor scenes with a 91.4% accuracy. We design six experiments based on types of input data (raw data, a full list of auditory features, and sliding windows with selected features) to investigate the effectiveness and efficiency of our SlideWin pipeline. The experiments shows that our SlideWin delivers superior performance of auditory scene detection, especially for Indoor / Outdoor scenes with 91.4% accuracy, and is stable and robust with different window sizes ranging from 14 to 99. The results based on our SlideWin will enhance practical applications of ML to analyze auditory scenes with limited annotated samples. It will further improve the recognition of environments that may potentially influence the safety of people, especially people with hearing aids and cochlear implant processors.

This work introduces a tool for quantifying and assessing recorded auditory scenes for an indoor or outdoor environment. While the data did support a comparison of auditory scenes between indoor and outdoor environments, limitations exist in the current analysis. One limitation is the sample size. Our analyses were based on a set of 200 auditory data collected in indoor and outdoor environment. However, our data currently only represent several locations in a few states. Collecting and analyzing data from a larger diverse and various settings of indoor and outdoor environment may allow us to generate additional findings and yield more robust scene prediction results.

A second limitation is the coarse granularity of the information that is provided by recorded scenes. These recorded data provide information on scenes in those settings. As a result, the captured features also indicate the characteristics of those settings. Including data form other types of environment can increase the diversity information that we analyze. For example, data from sensors (e.g., phones, smart watches) may provide insights on different environment settings that are useful for detecting environment and scene changes. In future work, we will investigate methods for predicting auditory scenes based on changes in an environment's features and / or characteristics. The results may provide timely and informed assistance or interventions to prevent and help with a variety of environmental related tasks (e.g., navigations).

Author Contributions:
Conceptualization, Theo Gueuret, Beiyu Lin, and Lingwei Chen; methodology, Theo Gueuret, Beiyu Lin, Lingwei Chen, Margaret McMullin, and Joel Synder; software, Theo Gueuret; validation, Theo Gueuret, Beiyu Lin, Margaret McMullin, and Joel Synder; formal analysis, Theo Gueuret, Beiyu Lin; investigation, Theo Gueuret, Beiyu Lin; resources, Joel Synder; data curation, Theo Gueuret, and Margaret McMullin; writing—original draft preparation, Beiyu Lin, and Lingwei Chen; methodology, Theo Gueuret, Beiyu Lin, Lingwei Chen, Margaret McMullin, and Joel Synder; writing—review and editing, Beiyu Lin, Lingwei Chen, Margaret McMullin, Ruben Alberts, Xiaowei Jia, and Joel Synder; visualization, Theo Gueuret, Beiyu Lin, and Lingwei Chen; supervision, Beiyu Lin; project administration, Beiyu Lin. All authors have read and agreed to the published version of the manuscript.", please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Data Availability Statement: This dataset will be publicly available when the article is published.

References

1. Barb, G.; Alexa, F.; Otesteanu, M. Dynamic spectrum sharing for future LTE-NR networks. *Sensors* **2021**, *21*, 4215.
2. Chandrakala, S.; Jayalakshmi, S. Environmental audio scene and sound event recognition for autonomous surveillance: A survey and comparative studies. *ACM Computing Surveys (CSUR)* **2019**, *52*, 1–34.
3. Tsiami, A.; Filntisis, P.P.; Efthymiou, N.; Koutras, P.; Potamianos, G.; Maragos, P. Far-field audio-visual scene perception of multi-party human-robot interaction for children and adults. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 6568–6572.
4. Do, H.M.; Sheng, W.; Liu, M.; Zhang, S. Context-aware sound event recognition for home service robots. In Proceedings of the 2016 IEEE International Conference on Automation Science and Engineering (CASE). IEEE, 2016, pp. 739–744.
5. Yasuda, M.; Ohishi, Y.; Saito, S.; Harado, N. Multi-View And Multi-Modal Event Detection Utilizing Transformer-Based Multi-Sensor Fusion. In Proceedings of the ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2022, pp. 4638–4642.
6. Mesaros, A.; Heittola, T.; Virtanen, T. TUT database for acoustic scene classification and sound event detection. In Proceedings of the 2016 24th European Signal Processing Conference (EUSIPCO). IEEE, 2016, pp. 1128–1132.
7. Heittola, T.; Mesaros, A.; Eronen, A.; Virtanen, T. Context-dependent sound event detection. *EURASIP Journal on Audio, Speech, and Music Processing* **2013**, *2013*, 1–13.
8. Chen, L.; Li, X.; Wu, D. Adversarially Reprogramming Pretrained Neural Networks for Data-limited and Cost-efficient Malware Detection. In Proceedings of the Proceedings of the 2022 SIAM International Conference on Data Mining (SDM). SIAM, 2022, pp. 693–701.
9. Ren, P.; Xiao, Y.; Chang, X.; Huang, P.Y.; Li, Z.; Gupta, B.B.; Chen, X.; Wang, X. A survey of deep active learning. *ACM computing surveys (CSUR)* **2021**, *54*, 1–40.
10. Hershey, S.; Chaudhuri, S.; Ellis, D.P.; Gemmeke, J.F.; Jansen, A.; Moore, R.C.; Plakal, M.; Platt, D.; Saurous, R.A.; Seybold, B.; et al. CNN architectures for large-scale audio classification. In Proceedings of the 2017 ieee international conference on acoustics, speech and signal processing (icassp). IEEE, 2017, pp. 131–135.
11. Farina, A. Human dimension of the soundscape: from individuals to society. In *Soundscape Ecology*; Springer, 2014; pp. 107–142.
12. Hall, D.A.; Irwin, A.; Edmondson-Jones, M.; Phillips, S.; Poxon, J.E. An exploratory evaluation of perceptual, psychoacoustic and acoustical properties of urban soundscapes. *Applied Acoustics* **2013**, *74*, 248–254.
13. Gygi, B.; Shafiro, V. From signal to substance and back: Insights from environmental sound research to auditory display design. In *Auditory display*; Springer, 2009; pp. 306–329.
14. Lemaitre, G.; Rocchesso, D. On the effectiveness of vocal imitations and verbal descriptions of sounds. *The Journal of the Acoustical Society of America* **2014**, *135*, 862–873.
15. Gygi, B.; Kidd, G.R.; Watson, C.S. Similarity and categorization of environmental sounds. *Perception & psychophysics* **2007**, *69*, 839–855.
16. McMullin, M.A.; Snyder, J.S. Dimensionality of natural auditory scene perception: A factor analysis study. *Psychonomic Society Annual Meeting* **2022**.
17. Snyder, J.S.; Elhilali, M. Recent advances in exploring the neural underpinnings of auditory scene perception. *Annals of the New York Academy of Sciences* **2017**, *1396*, 39–55.
18. Mlynarski, W.; McDermott, J.H. Ecological origins of perceptual grouping principles in the auditory system. *Proceedings of the National Academy of Sciences* **2019**, *116*, 25355–25364.
19. Basbug, A.M.; Sert, M. Acoustic scene classification using spatial pyramid pooling with convolutional neural networks. In Proceedings of the 2019 IEEE 13th International Conference on Semantic Computing (ICSC). IEEE, 2019, pp. 128–131.
20. Fedorishin, D.; Sankaran, N.; Mohan, D.D.; Birgiolas, J.; Schneider, P.; Setlur, S.; Govindaraju, V. Waveforms and Spectrograms: Enhancing Acoustic Scene Classification Using Multimodal Feature Fusion. In Proceedings of the DCASE, 2021, pp. 216–220.
21. Koutini, K.; Eghbal-Zadeh, H.; Dorfer, M.; Widmer, G. The receptive field as a regularizer in deep convolutional neural networks for acoustic scene classification. In Proceedings of the 2019 27th European signal processing conference (EUSIPCO). IEEE, 2019, pp. 1–5.
22. Yin, Y.; Shah, R.R.; Zimmermann, R. Learning and fusing multimodal deep features for acoustic scene categorization. In Proceedings of the Proceedings of the 26th ACM international conference on Multimedia, 2018, pp. 1892–1900.
23. Weiping, Z.; Jiantao, Y.; Xiaotao, X.; Xiangtao, L.; Shaohu, P. Acoustic scene classification using deep convolutional neural network and multiple spectrograms fusion. *Detection and Classification of Acoustic Scenes and Events (DCASE)* **2017**.
24. Bregman, A.S. *Auditory scene analysis: The perceptual organization of sound*; MIT press, 1994.