Improving beehive state prediction from audio using relevant audio timbre features and deep neural networks

Khellouf Leila
Intership tutor: Dominique Fourer, Dominique Cassou-Ribehart, Jean-Paul Gavini

September 2020

Content

- Introduction
- Some key concepts
- Proposed Approach
- Feature extraction
- Classification
- Feature Selection
- Evaluation and Results
- Conclusion and Future works

Introduction

Honey bee is one of the most important pollinator insects for flowers, fruits and vegetables and they are well known for their positive effects but in recent years multiple stress factors have led to a decline of honey bee colonies.

Related problem focus on natural audio recordings extracted from a unique beehive.

The objective of this inter-ship is to detect the state of the health of the beehive using audio signal as input data.

Process of the project

The steps of this project are:

- We investigated the most relevant audio features which enable to efficiently monitor a beehive from noisy field recordings.
- Build machine learning models to classify, describe, and generate audio.
- We will develop and assessment of one or several new techniques using natural recordings provided by **Starling Partners company**

Some Concepts

What is Audio ?

An audio signal is a representation of sound, typically as an electrical voltage. Audio signals have frequencies in the audio frequency range of roughly 20 to 20,000 Hz (the limits of human hearing). There types of signal domain are:

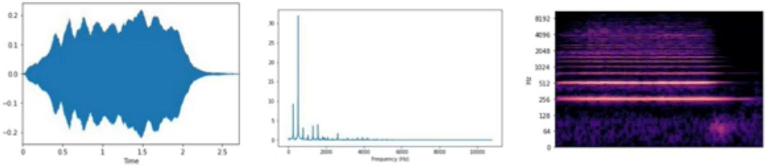


Figure: Signal domain

Some Concepts

Fourier Transform & Spectogram

Fourier Transform

Fourier Transform is a tool to transform a wave function or signal from a time domain into frequency domain.

Spectrogram

A spectrogram is a visual representation of the spectrum of frequencies of a signal as it varies with time.

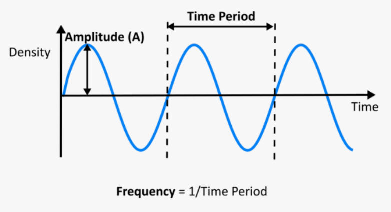


Figure: Amplitude, Time Perdiod and Frequency of Sound Wave

Data Exploratory

The dataset [1] used in this project was developed in the context of the work in the paper [2] that focuses on the automatic recognition of beehive sounds. The annotated dataset was developed based on a selected set of recordings acquired in the context of two different projects:

- The Open Source Beehive (OSBH).
- The NU-Hive project

[1] 1https://zenodo.org/record/2563940.XGVwpDP7SUk

[2] Ines Nolasco2, Alessandro Terenzi1, Stefania Cecchi1, Simone Orcioni1, Helen L. Bear2, and Emmanouil Benetos ,AUDIO-BASED IDENTIFICATION OF BEEHIVE STATES

Dataset

As reported in the paper [3] the data was acquired continuously with:

- Fs= 22.05 khz
- Microphones are MEMS type.
- Time segments are labled as Bee or Nobee.
- We have 78 recordings of varying lengths.

[3] Ines Nolasco2, Alessandro Terenzi1, Stefania Cecchi1, Simone Orcioni1, Helen L. Bear2, and Emmanouil Benetos ,AUDIO-BASED IDENTIFICATION OF BEEHIVE STATES

Dataset

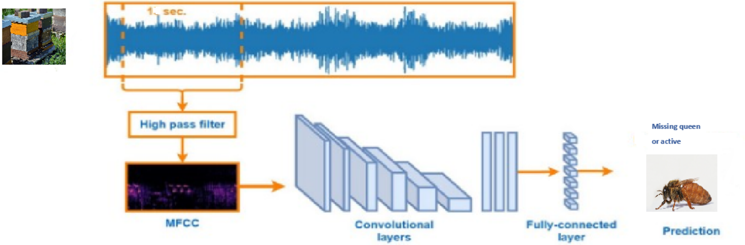
In our case, we only used six hives , 2 hives from OSBH project and 4 hives from NU-Hive project. we decompose each audio signal into one second. The table 1 show the decomposition of the data by hive (e.g. CF003 is made of 3700 s of audio signal labeled as active− bee)

State\ Hive	Label	CF001	CF003	*CJ*001	*GH*001	Hive1	Hive3
Active	bee	0	3700	0	1401	2687	656
	nobee	0	500	0	1724	901	530
Missing Queen	bee	16	0	802	0	1476	6557
	nobee	0	0	698	0	903	2265

Table: Content of the investigated database expressed in seconds

Illustration method:

The proposed approach is based on two steps. Firstly, feature extraction with the aim of determining the frequency behaviour of the beehive when the queen is present or not. Secondly, classification of beehive states.



The solution summary — audio data preprocessing and neural networks model

Feature Extraction

What is Feature Extraction ?

Feature extraction is the process of computing a compact numerical representation that can be used to characterize a segment of audio.

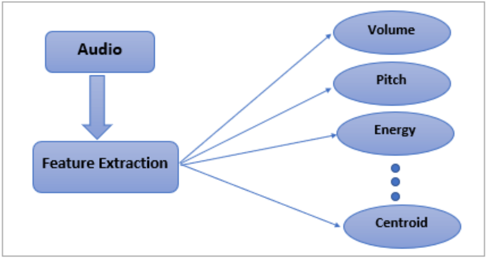
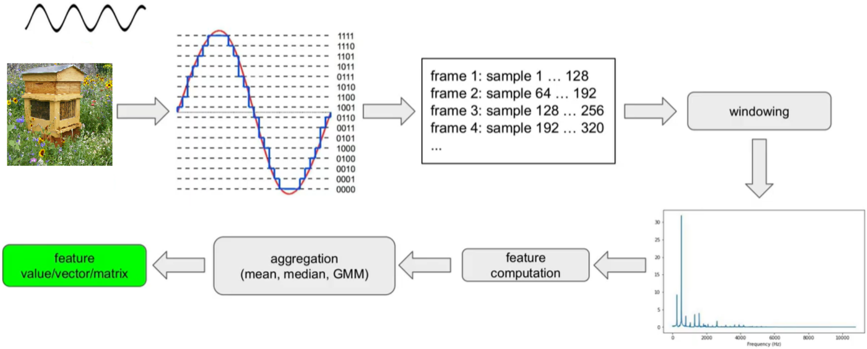


Figure: Process of the feature selection

Feature Extraction

How do we extract audio features ?



frame 1: sample 1 ... 128
frame 2: sample 64 ... 192
frame 3: sample 128 ... 256
frame 4: sample 192 ... 320
...

windowing

feature computation

aggregation (mean, median, GMM)

feature value/vector/matrix

Figure: The procedure for extracting the audio features

In this project we used three techniques of feature extraction:

Mel-frequency cepstral coefficients

MFCCs were introduced by Davis and Mermelstein in the 1980's, and have been state-of-the-art ever since. It is based on the non linear mel frequency scale which finds a large number of audio applications such as speech recognition and music information retrieval

Feature Extraction

MFCC Vs STFT Vs Timbre audio toolbox

The Short-Time Fourier Transform (STFT)

The STFT is a list of linear transformations of functions related to Fourier analysis. The procedure for computing STFTs is to divide a longer time signal into shorter segments of equal length and then compute the Fourier transform separately on each shorter segment.

Audio Timbre Toolbox

The Timbre Toolbox provides a comprehensive set of descriptors that can be useful in perceptual research, as well as in music information retrieval and machine-learning approaches to content-based retrieval in large sound databases

Audio Classification

To classify the audio into Queen or Missing Queen. We investigates three distinct Supervised classification methods which are :

- Suport Vector Machines (SVM)
- Convolutional Neural Network (CNN)
- Dense Neural Network (DNN)

In this project, we use a Gaussian radial basis function $f(x, \hat{x}) = \exp\left(-\gamma ||x - \hat{x}||^2\right)$ where γ is defined as one divided by the number of input features. In our study, we used the python implementation provided by Scikit-learn[4].

[4] https://scikit-learn.org

Audio Classification

CNN architecture

The size of the input layer depends on the choice of the computed audio features ((880) for MFCC and $(22, 528)$ for STFT).

#	Output size	Layer type	Filter size
1	$20 \times 44 \times 1$	Input	
2	$20 \times 44 \times 16$	Convolutional	3×3
3	$20 \times 44 \times 16$	BatchNormalization	-
4	$10 \times 22 \times 16$	Max pooling	2×2
5	$10 \times 22 \times 16$	Convolutional	3×3
6	$10 \times 22 \times 16$	BatchNormalization	-
7	$5 \times 11 \times 16$	Max pooling	2×2
8	$5 \times 11 \times 16$	Convolutional	3×1
9	$5 \times 11 \times 16$	BatchNormalization	-
10	$3 \times 6 \times 16$	Max pooling	2×2
11	$3 \times 6 \times 16$	Convolutional	3×1
12	$3 \times 6 \times 16$	BatchNormalization	-
13	$2 \times 3 \times 16$	Max pooling	2×2
14	256	Fully connected	
15	32	Fully connected	
16	2	Fully connected	
17	2	Soft max	
18	1	Output	

Table: Architecture of our proposed CNN.

Audio Classification

DNN architecture

The input layer of the DNN corresponds to the 164 timbre feature coefficients.

#	Output size	Layer type
1	164 × 1	Input
2	164	Fully connected
3	328	BatchNormalization
4	328	Fully connected
5	328	BatchNormalization
6	328	Fully connected
7	328	BatchNormalization
17	2	Soft max
18	1	Output

Table: Architecture of the proposed DNN for which the input layer corresponds to the 164 computed timbre features.

In in both experiments we used the RMSprop optimizer with a batch size equal to 145 and a number of epochs of 50.

Feature Selection

Mutual Information

Features selection algorithms aim at computing a subset of descriptors that conveys the maximal amount of information to model classes

Mutual information

The mutual information (MI) is a measure of the amount of information that one random variable has about another variable. The relevance can be measured with the mutual information defined by:

$$I(C, F) = \sum_{c} \sum_{f} P(c, f) \log \frac{P(c, f)}{P(c)P(f)} \tag{1}$$

PCA

PCA is fundamentally a simple dimensionality reduction technique that transforms the columns of a dataset into a new set features.It does this by finding a new set of directions (like X and Y axes) that explain the maximum variability in the data.

Results

As a pre-processing step, the audio signals are segmented in one-second-long homogeneous time frames (with the same label) and are resampled at $F_s = 22.05$ kHz to obtain 17,295 distincts individual where 8,444 are labeled as “active” and 8,851 are labeled as “missing queen”. Two distinct experimental protocols are used to comparatively assess the different investigated methods.

Results

Random Split

We used a random split of the whole dataset and merges the data from the 6 beehives. This configuration uses 70% of the dataset as a training set and 30% as the test set.

Method	Size	Label	F-measure	Prec.	Rec.	Acc.
MFCC+SVM	$\mathbb{R}^{20\times44}$	missing-queen	0.91	0.87	0.96	0.90
		active	0.90	0.85	0.90	
MFCC+CNN	$\mathbb{R}^{20\times44}$	missing-queen	0.99	1	0.99	0.99
		active	0.99	0.99	1	
STFT+CNN	$\mathbb{R}^{512\times43}$	missing-queen	0.30	0.97	0.18	0.58
		active	0.70	0.54	0.99	
Timbre feat.+SVM	$\mathbb{R}^{164}$	missing-queen	0.87	0.82	0.93	0.86
		active	0.85	0.92	0.79	
Timbre feat.+DNN	$\mathbb{R}^{164}$	missing-queen	0.90	0.87	0.94	0.90
		active	0.89	0.94	0.85	

Table: Comparative results obtained for the random split experiment.

Results

4-Fold Cross Validation

The second experiment uses a 4-fold-cross validation where each fold corresponds to a distinct beehive except for four hives which are merged into two distinct folds (CF001+CF003 and CJ001+GH001) because they only contain "active" (resp. "missing queen" individuals).

Fold\ De	Training	Test
Fold 1	$CJ001 + GH001 + Hive1 + Hive3$	$CF001 + CF003$
Fold 2	$CF001 + CF003 + Hive1 + Hive3$	$CJ001 + GH001$
Fold 3	$CF001 + CF003 + CJ001 + GH001 + Hive3$	$Hive1$
Fold 4	$CF001 + CF003 + CJ001 + GH001 + Hive1$	$Hive3$

Table: Proposed manual partitioning of the data-set to apply a $4-$fold$-$cross validation

Results

Interestingly, the 4-fold-cross-validation experiment shown in the table bellow

Method	Size	Label	F-measure	Prec.	Rec.	Acc.
MFCC+SVM	$\mathbb{R}^{20\times 44}$	missing-queen	0.12	0.09	0.45	0.09
		active	0.01	0.01	0.01	
MFCC+CNN	$\mathbb{R}^{20\times 44}$	missing-queen	0.34	0.27	0.51	0.41
		active	0.38	0.417	0.57	
STFT+CNN	$\mathbb{R}^{512\times 43}$	missing-queen	0.003	0.33	0.021	0.61
		active	0.76	0.87	0.86	
Timbre feat.+SVM	$\mathbb{R}^{164}$	missing-queen	0.25	0.25	0.39	0.31
		active	0.33	0.38	0.33	
Timbre feat.+DNN	$\mathbb{R}^{164}$	missing-queen	0.30	0.31	0.38	0.45
		active	0.54	0.6	0.7	

Table: Comparative results obtained for the 4-fold cross-validation experiment.

Results

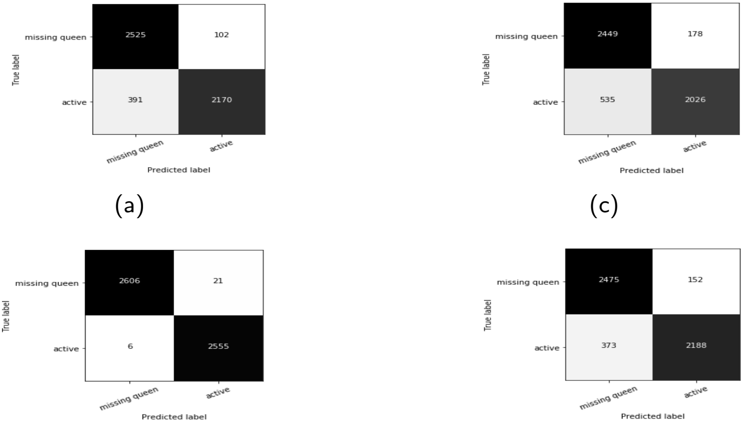
In both experiment, the timbre features obtains the most balanced results with a significant lower number of parameters (164 real scalars) in comparison to MFCC (880) and STFT (22,528). This clearly shows an advantage of the timbre features to obtain good prediction using a very low number of computed features. The dataset[5] and python code [6] developed for this work are publicly available.

[5] *https : //zenodo.org/record/*2563940.*XGVwpDP7SUk*

[6] *https : //github.com/khelloufleila/AUDIO − BASED − IDENTIFICATION − OF − BEEHIVE − STATES*1

Results

Confusion Matrix



(a) (c)

(b) (d)

Figure: Confusion matrices obtained respectively with the MFCC+SVM (a), MFCC+CNN (b), Timbre feat.+SVM (c) and Timbre feat.+DNN (d) for the

Results

Feature selection

We use mutual information theory to select a subset of features from an original feature set based on feature-class information values. We evaluate our feature selection method using classification method.

The different steps used are:

- Calculate the MI of the 164 features.
- Rank the score of features in decreasing order of the relevance
- Calculate F1-score by using Logistic regression algorithm.

Results

Feature selection

As the inflection point shown in Figure bellow, with a minimum of features (i.e 82 features), we get a maximum score.

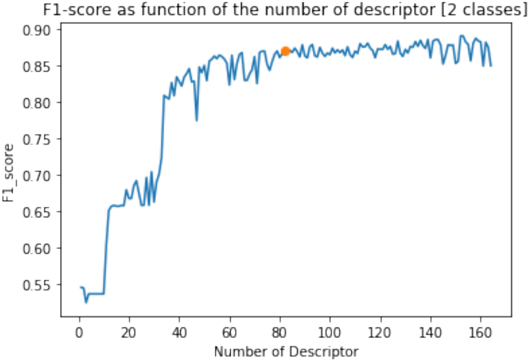


Figure: F1-score as function of the number of descriptor (timbre toolbox + LogisticRegression)

Results

PCA for MFCC

In the MFCC Scatterplot, though there is a certain degree of overlap, the points in MFCC Scatterplot belonging to the same category are distinctly clustered for each Hive and region bound of the 2 classes.

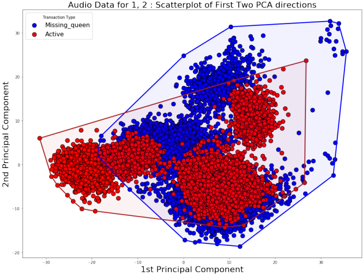


Figure: Audio data for Missing queen and Active : Scatterplot of First Two PCA directions for MFCC

Results

PCA for Audio Timber Toolbox

In the other hand, the points in Audio Timbre Scatterplot belonging to the same category are distinctly clustered and region bound.

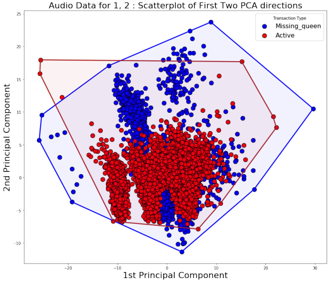


Figure: Audio data for Missing queen and Active : Scatterplot of First Two PCA directions for Audio Timber Toolbox .

Conclusion

The audio timbre toolbox shows an improvement of the model robustness when compared to several state-of-the-art methods. It allows to significantly reduce the number of required input size to 164 real scalars and paves the way of efficient embedded-system-based implementations. However, the cross-hive analysis results appear to be insufficient and will require a further investigation involving a larger annotated dataset.

Future works

Future work will consist in a real-world integration of our approach using an embedded system and a consideration of more beehive state labels such as bees swarming or queen piping and quacking which are full of interest for beekeepers.

Thank You For Your Attention