Audio Event Detection for Automatic Scene Recognition

Xun Xu

Department of Computer Science and Engineering
Shanghai Jiao Tong University

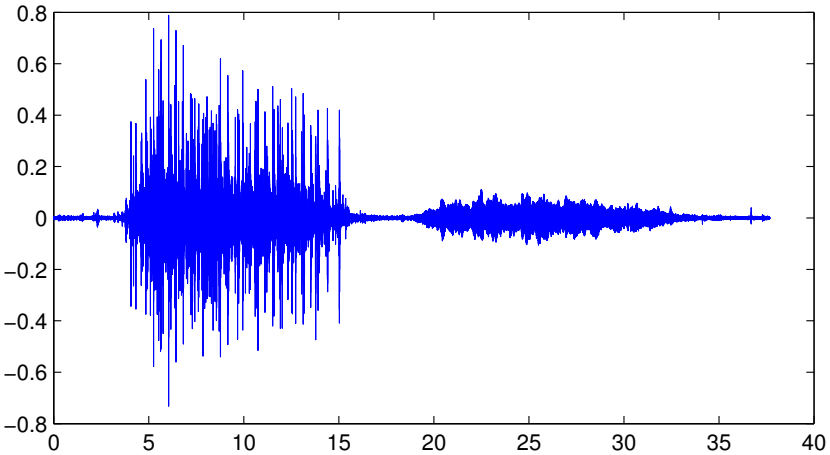
June 22, 2015

Outline

Outline

Problem Description

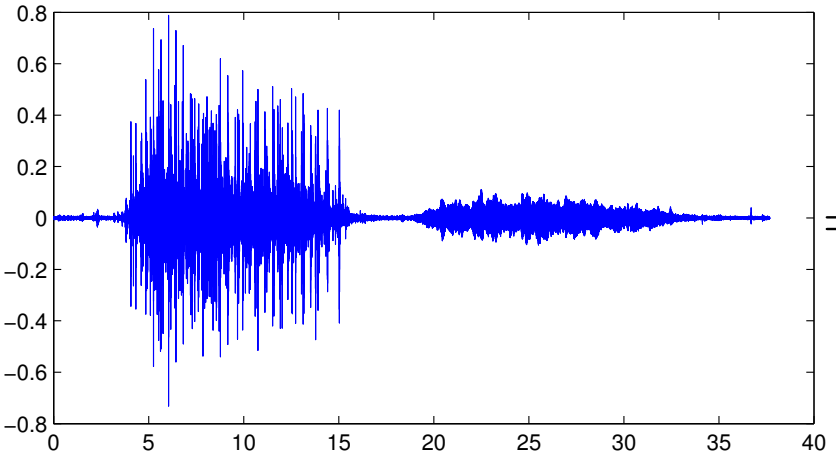
In this project, our problem is to recognize a scene where an audio is recorded.



$\Rightarrow$ *concert*

Our Approach

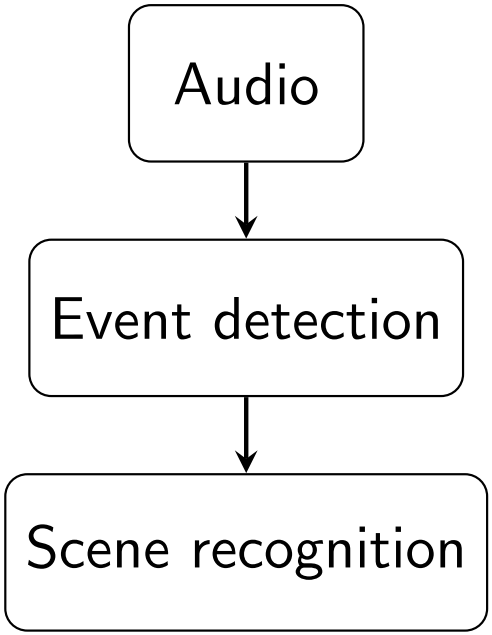
Our approach is to detect the audible events in a clip.
Then infer the scene from the detected events.



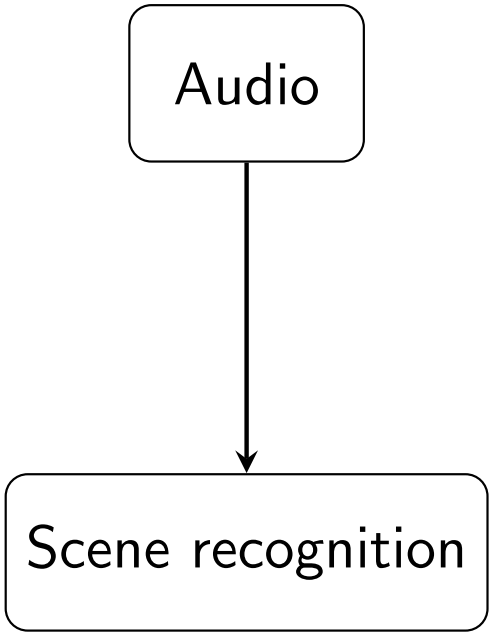
$\Rightarrow$ *applause, instrument* $\Rightarrow$ *concert*

Our Approach vs. Other Approaches

Our approach:



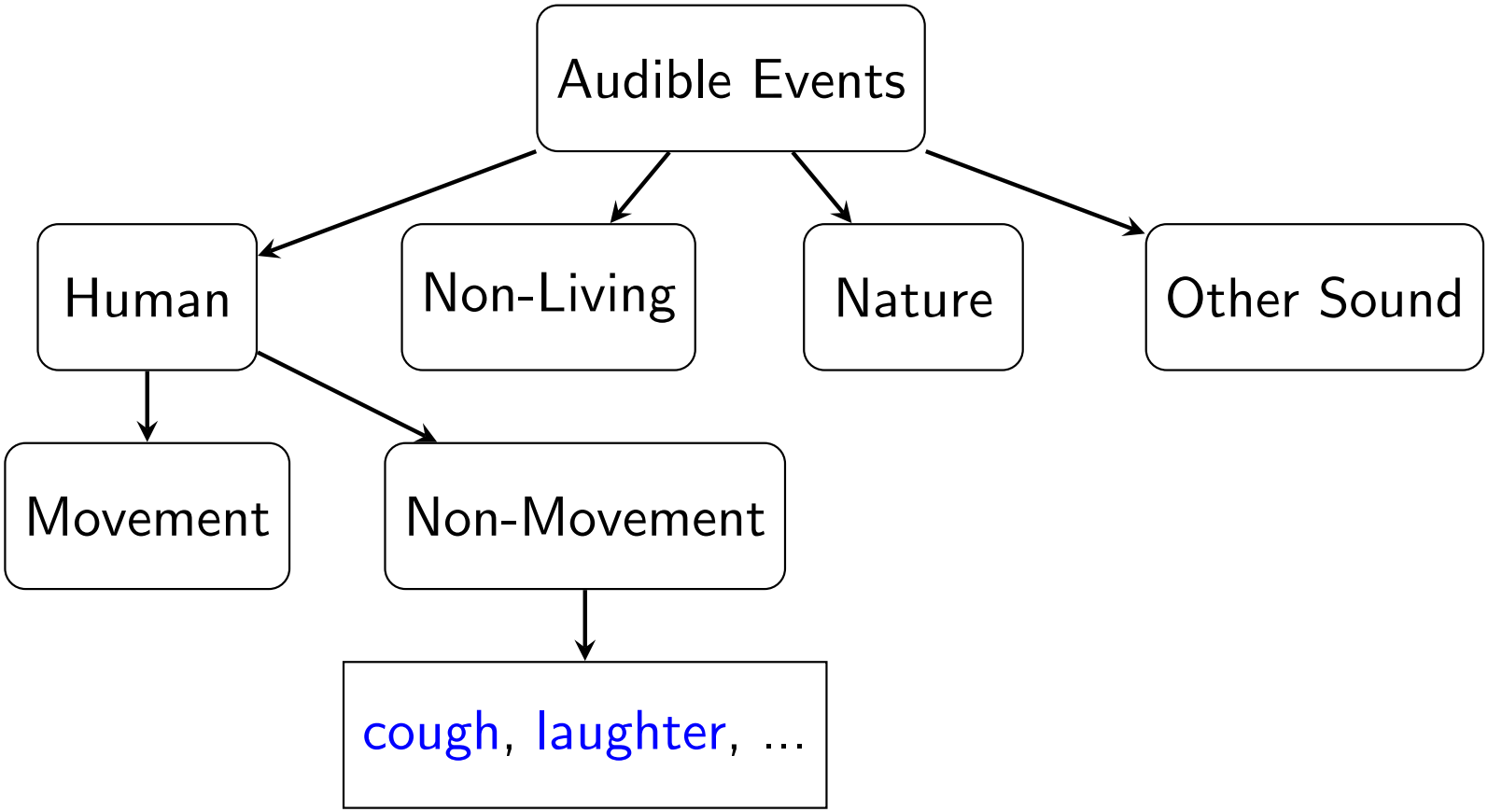
Other approaches:



Outline

Audible Event Taxonomy

We labelled common audible events into 4 classes.
There are 120 events in total.



Audio Data

We download the audio data for events from Sound Search Engines (SSEs).
For example, when we query "cough" in SSE:

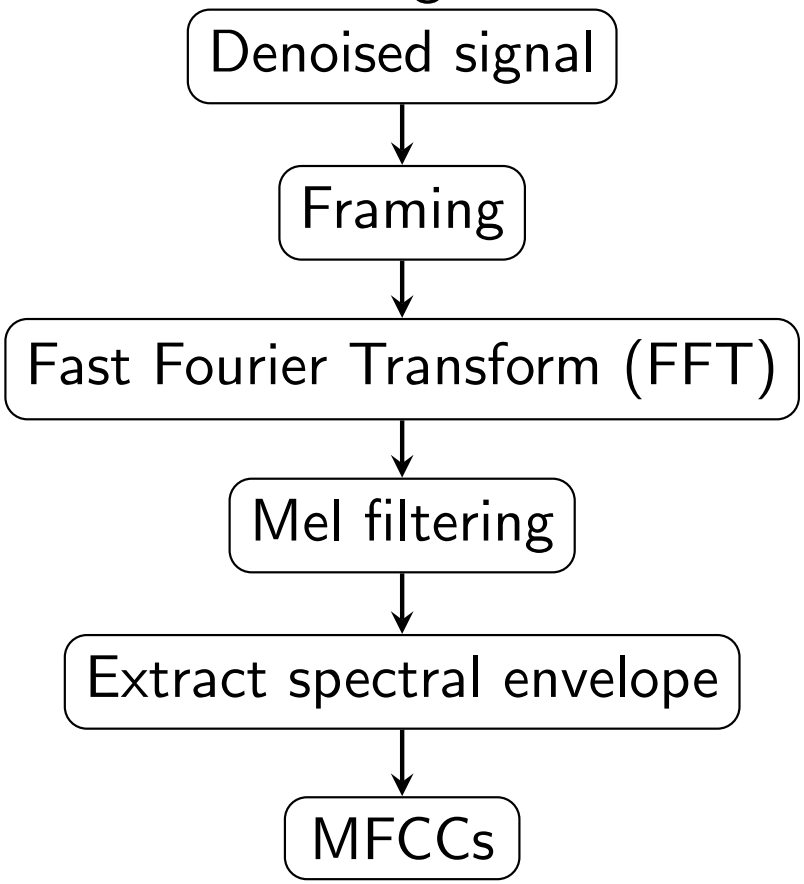
We download clips from 1 second to 60 seconds.

Preprocess and Feature Extraction

We first use first use Minimum Statistics to calculate the noise spectrum and subtract it from the input signal.
Then we extract Mel-Frequency Cepstral Coefficients (MFCCs) from denoised signal.

Preprocess and Feature Extraction

We first use first use Minimum Statistics to calculate the noise spectrum and subtract it from the input signal.
Then we extract Mel-Frequency Cepstral Coefficients (MFCCs) from denoised signal.



Event Model

We use features to train Gaussian Mixture Models (GMMs).
The training is done by Expectation-Maximization (EM) algorithm.

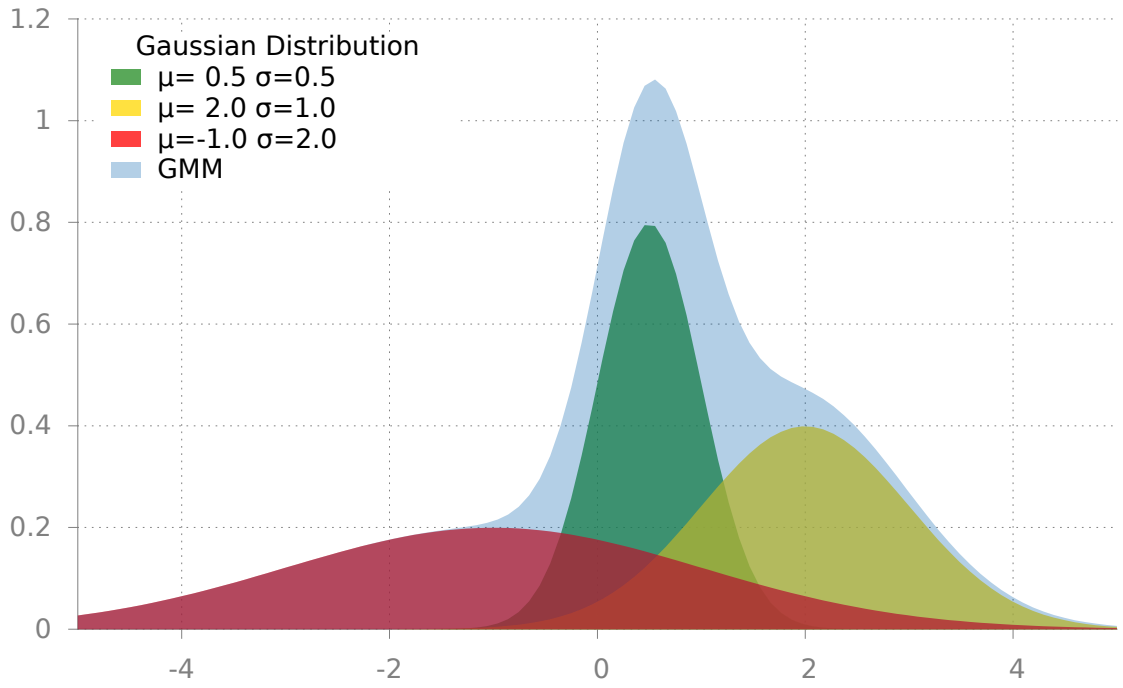


Figure: A GMM with three components

Outline

Scene Extraction

We use the scripts for movies, plays and TV series to extract the scenes.
Below is a script example. We call it a *context*, including a scene, and some descriptive sentences.

INT. LEONARD'S BATHROOM - Night
Leonard turns on the light, revealing a shower, toilet and sink.
He removes toiletries from the grocery bag and places them inside.

Scene Extraction

We use Natural Language Process (NLP) tools to process a context, and eliminate the following type of words:

- Person names
- Time indicator
- Location names
- Adjective, determiner, number, ...

Table: Top 10 Occurred Scenes

Scene	Occurrence
house	3537
office	3259
apartment	2919
room	2580
bedroom	2257
car	1699
street	1622
kitchen	1431
living room	1374
tardis	1259

Scene-Event Relation Mining

To get the relation between scenes and audible events,
we match the context in a script with our predefined audible events.

INT. LEONARD'S BATHROOM - Night
Leonard turns on the light, revealing a shower, toilet and sink.
He removes toiletries from the grocery bag and places them inside.

Scene-Event Relation Mining

Based on the idea of Term-Frequency-Inverse Document Frequency (TFIDF), we calculate two scores of an event e, to a scene s.

1. $TF = log(1 + f(e, s))$
   $f(e, s)$ is the number of contexts e appears in all contexts under scene s.
2. $IDF = 1 + log(\frac{N}{N_e})$
   N is the number of scenes. N_e is the number of scenes in which event e appears.

These two scores are then multiplied, and used as the importance of an event to a scene.

$$TFIDF = TF \times IDF \tag{1}$$

Table: An example of scene-event map

Scene	Top 10 events ranked by TF-IDf
bathroom	running+water, toilet, faucet, toothbrush, shower, drawer, drain, talk, paper, bowl
beach	seagull, sand, boat, talk, wave, sea, car, laughter, drink, wood, running
concert	piano, applause, crowd, chorus, child, cry, talk
forest	tree, wood, dirt, talk, running, bird, river, car, leaf, grass, wind
kitchen	drawer, cutlery, microwave, dish, kettle, talk, bowl, phone, toaster, running+water
office	desk, drawer, page+turn, talk, phone, printer, paper, chair, leaf, typewriter
park	talk, car, tree, laughter, dog, child, grass, crowd, running, phone
restaurant	talk, drink, laughter, phone, car, leaf, paper, dish, ring, chair, write
street	car, truck, subway, talk, traffic, engine, siren, phone, running, laughter
subway station	subway, train, car, tube, talk, pace, crowd, metal, phone, vehicle

Audio Segmentation

In testing, we segment the audio into smaller parts for event detection. We set two thresholds based on the following two features:

1. Frame Energy
   The averaged energy of a frame, calculated as:

$$E_i = \frac{\sum_{n=1}^{N} (x_i(n))}{N} \tag{2}$$

2. Spectral Centroid
   The "center" of frequency, calculated as:

$$C_i = \frac{\sum_{k=1}^{N} k \times Amp(k)}{\sum_{k=1}^{N} Amp(k)} \tag{3}$$

Audio Segmentation

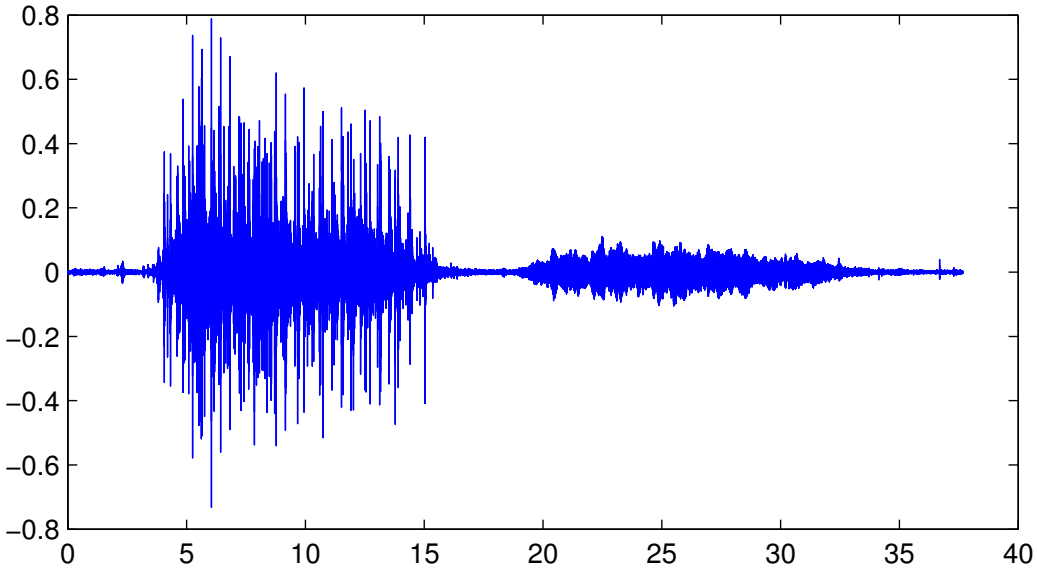


Figure: A example audio clip

Audio Segmentation

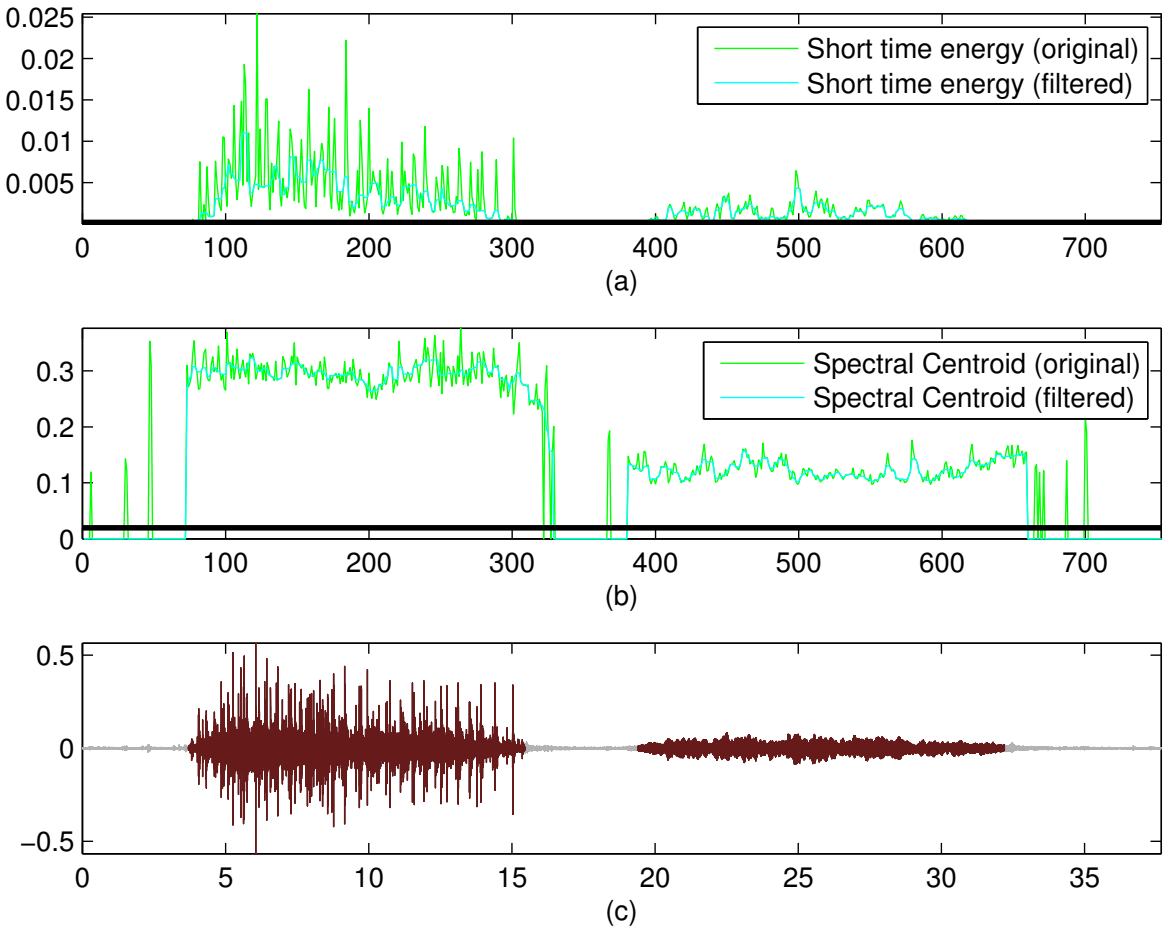
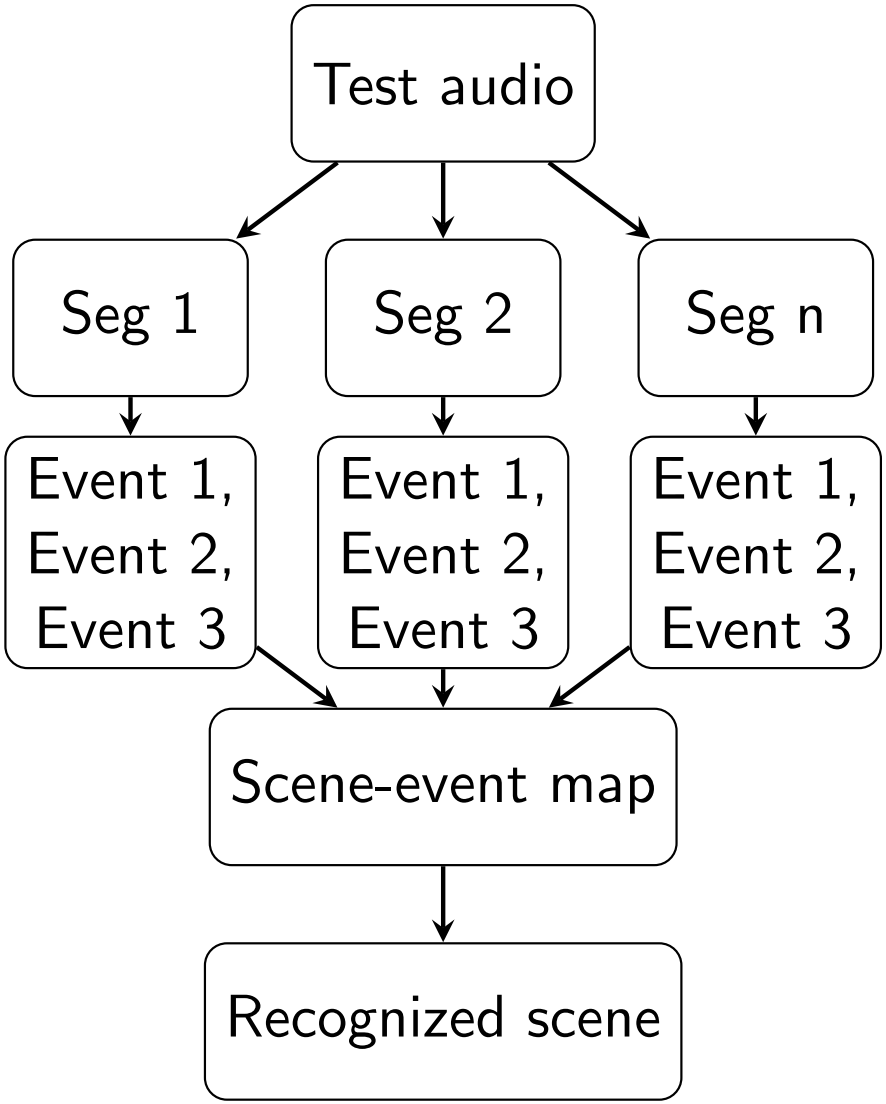


Figure: A segmentation example

Scene Inference

For each segment, we evaluate it with our trained GMMs.
We choose the top three detected events for scene voting.



Outline

Component Number Evaluation

Gaussian Mixture Model distribution:

$$P(\mathbf{x}|\pi,\mu,\Sigma) = \sum_{k=1}^{M} \pi_k \mathcal{N}(\mathbf{x}|\mu_k,\Sigma_k), \tag{4}$$

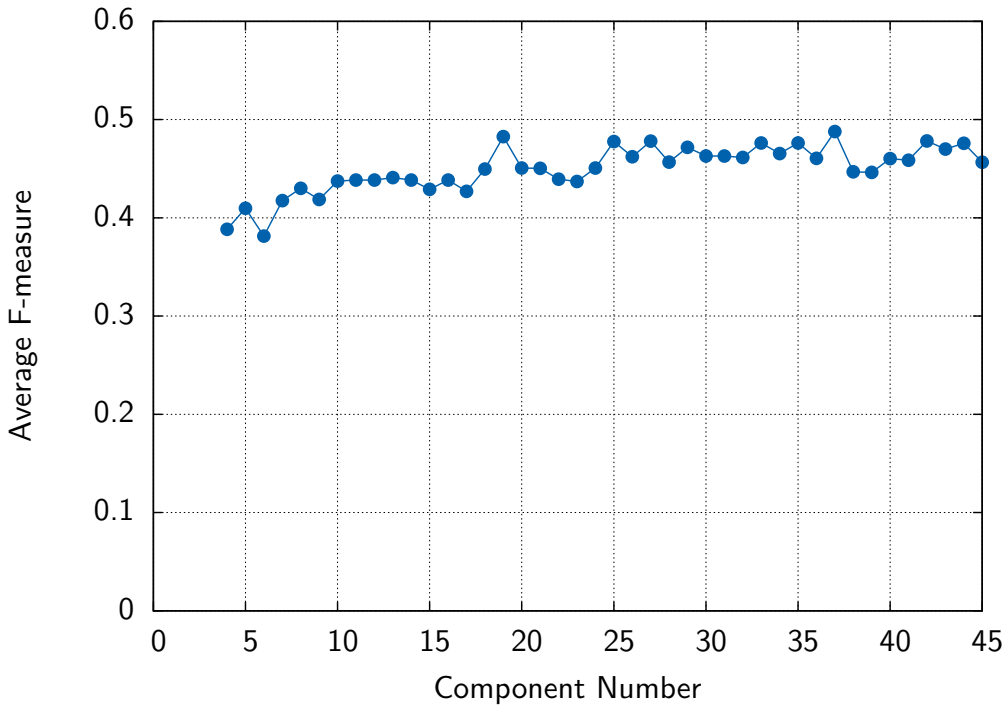


Figure: F-measure for different component number

Componnent Number Evaluation

After comparing F-measure and running time, we choose 18 as our component number.

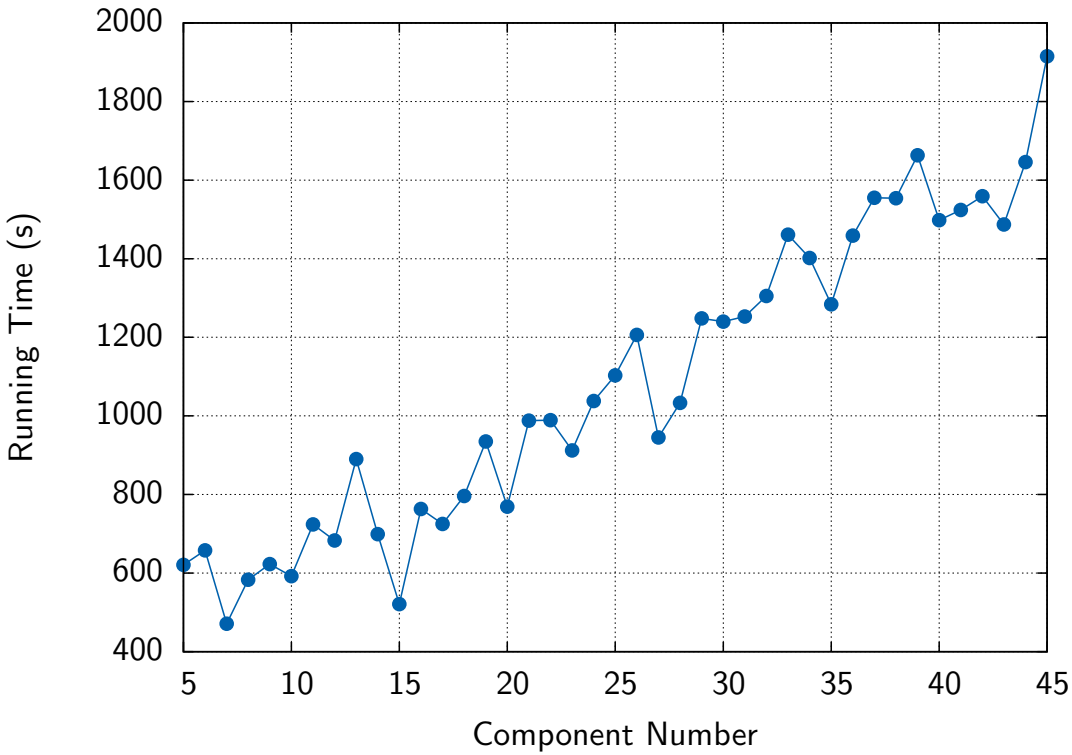


Figure: Running time for different component number

Event Detection Evaluation

A event detection task of 16 events.
The other six systems are chosen from IEEE AASP Challenge.

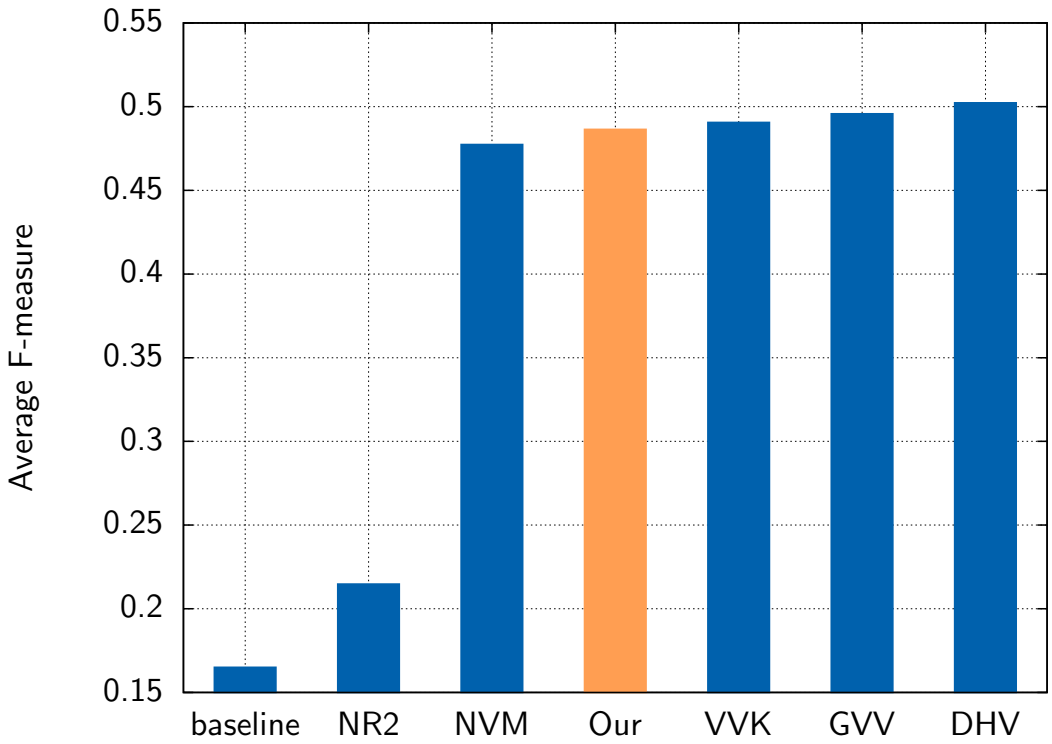


Figure: Event Detection F-Measure

Scene Recognition Evaluation

In scene recognition, we choose 10 scenes, each scene has 10 clips. Accuracy for other 4 systems are calculated using 5-fold cross validation.

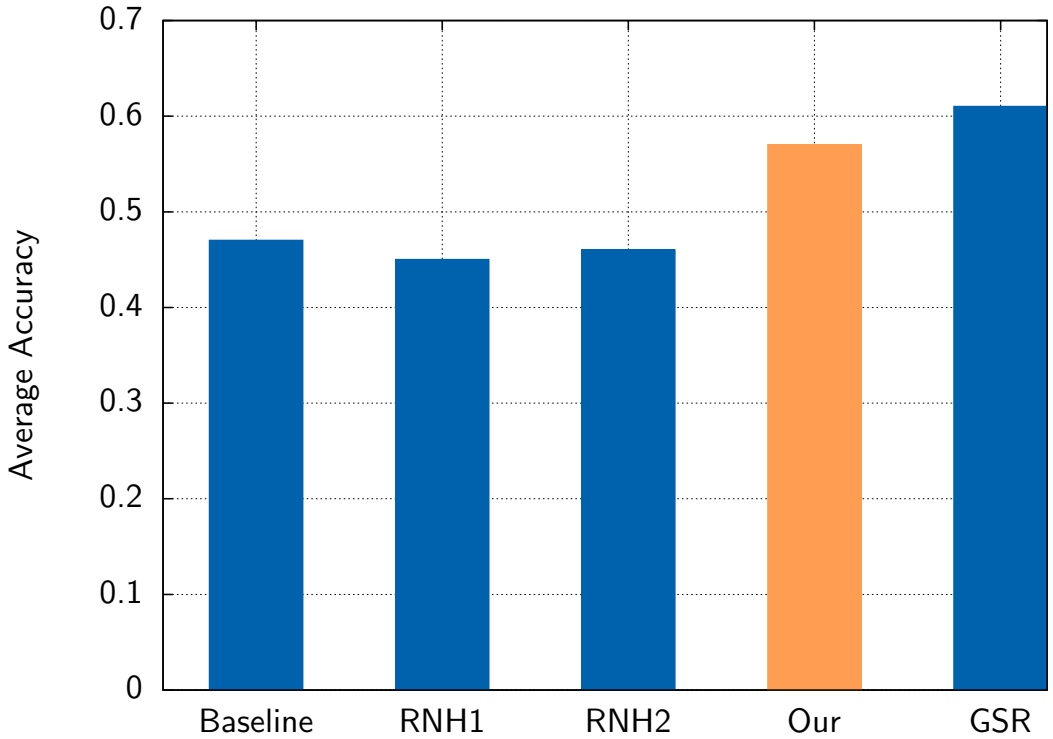


Figure: Recognition accuracy for 10 audio scenes

Scene Recognition Evaluation

Detailed result of our system with the best system *GSR*.

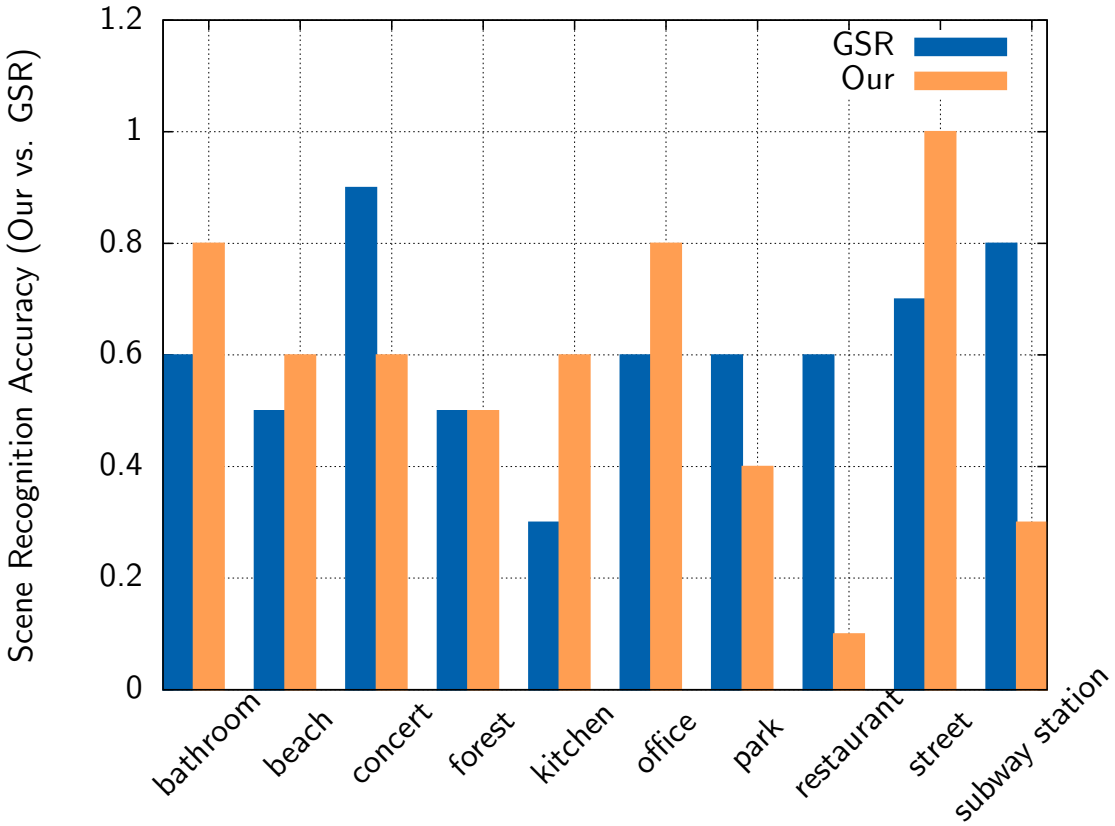


Figure: 10 scenes comparison

Outline

Demo

Live demo for our system.

Acknowledgement

I would like to thank my advisor Kenny Q. Zhu, and my friend Xinyu Hua for their help in this project.

Thank you!

Any Question?