SOMPHONY: Visualizing Symphonies using 3D Self-Organizing Maps

Cruz, Edwardo
Dionisio, Jefferson
Fukuoka, Kenji
Portales, Naomi

Advised by: Fritz Kevin Flores

PRESENTATION OUTLINE

1. Research Description
    a. Introduction
    b. Research Gap
    c. Research Objectives with respective scopes and limitations
    d. Research Significance
2. Research Methodology
    a. Research Activities
    b. Calendar of Activities

Introduction

Musical Eras

Baroque Period (1600 to 1750)

Classical Period (1750 to 1820)

19th Century (1814-1914)

Romantic Period (1830-1910)

20th Century (1900-2000)

A History of Western Music (Grout, D., Palisca, C. 1996)
Nineteenth-Century Music (Dahlhaus, C. 1989)

Review of Related Literature

A Survey on Symbolic Data-Based Music Genre Classification

- Corrêa, D. C., & Rodrigues, F. A., 2016
- Ever-expanding music database
- hard to classify music genre
- Symbolic-based music feature used for training system for genre classification (MIDI, KERN)

Automated Motivic Analysis via Melodic Clustering

- Cambouropoulos, E. and Widmer, G. (2000)
- Finding similarity in music patterns
- Use differences in pitch-intervals and rhythm as basis for splitting one musical motive (small bits of music) from another

Validating the Stable Clustering of Songs in a Structured 3D SOM

- Azcarraga, A., Caronongan, A., Setiono, R., & Manalili, S. (2016)
- Construct 2D SOM as 3D SOM using similar learning algorithm (cube)
- Pre-processing (learning and labelling algorithm) and construct into a cube

SOMphony: Visualizing Symphonies Using Self-Organizing Maps

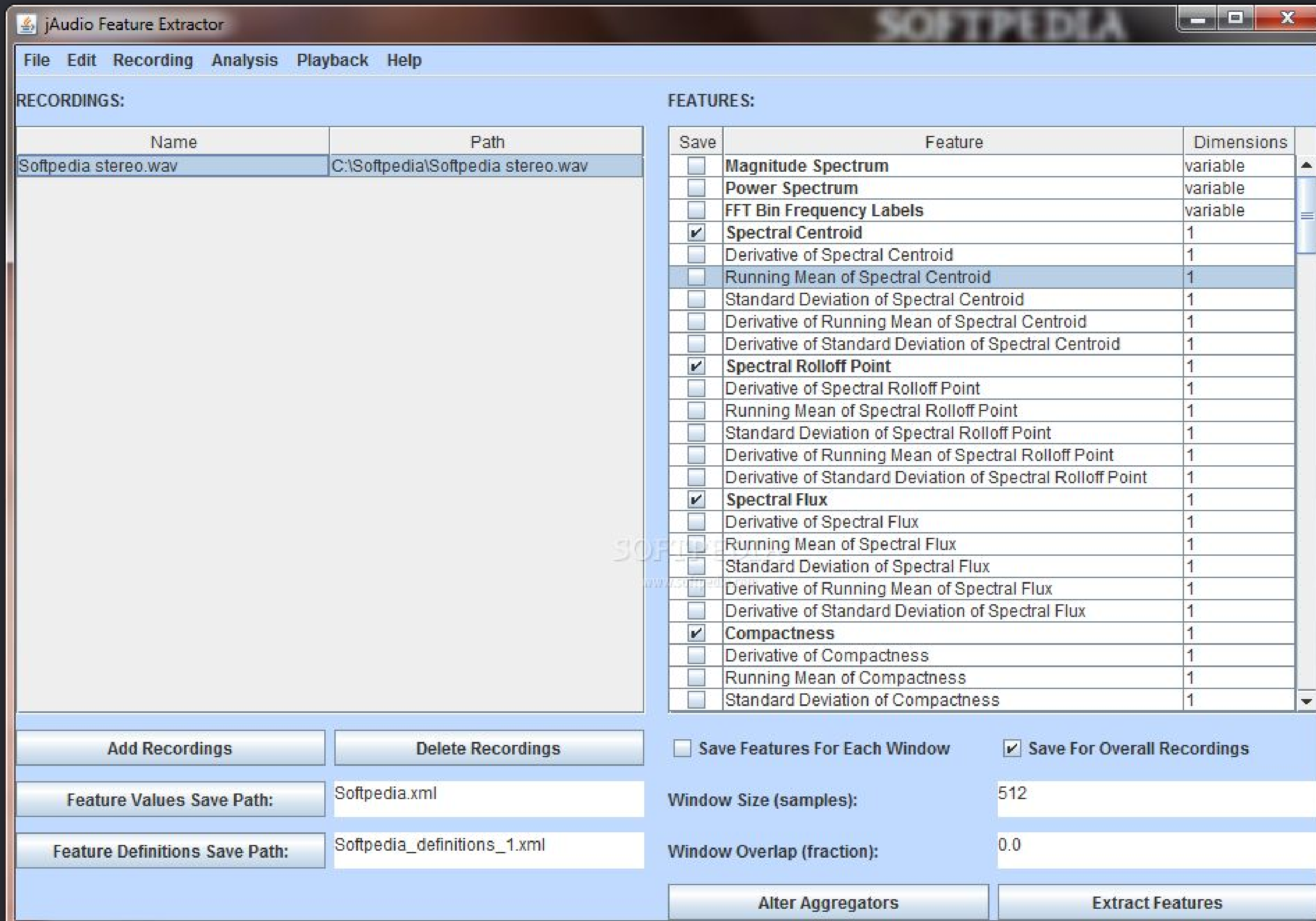
- Azcarraga & Flores (2016)
- Influence of composers to others
- Compare using 2D SOMs to find similarity among symphonies

SOMphony: Visualizing Symphonies Using Self-Organizing Maps

Makes use of **jAudio** for audio feature extraction and feature is used to feed data into machine learning algorithms.

Self-Oragnizing Maps (SOMs) are used to encode the musical trajectory of the different symphonies for visual analysis

K-means Clustering is used to partition similar nodes from the SOM Map


jAudio Feature Extractor
File
Edit
Recording
Analysis
Playback
Help
RECORDINGS:
Name
Path
Softpedia stereo.wav
C:\Softpedia\Softpedia stereo.wav
FEATURES:
Save
Feature
Dimensions
Magnitude Spectrum
variable
Power Spectrum
variable
FFT Bin Frequency Labels
variable
Spectral Centroid
1
Derivative of Spectral Centroid
Running Mean of Spectral Centroid
Standard Deviation of Spectral Centroid
Derivative of Running Mean of Spectral Centroid
Derivative of Standard Deviation of Spectral Centroid
Spectral Rolloff Point
Derivative of Spectral Rolloff Point
Running Mean of Spectral Rolloff Point
Standard Deviation of Spectral Rolloff Point
Derivative of Running Mean of Spectral Rolloff Point
Derivative of Standard Deviation of Spectral Rolloff Point
Spectral Flux
Derivative of Spectral Flux
Running Mean of Spectral Flux
Standard Deviation of Spectral Flux
Derivative of Running Mean of Spectral Flux
Derivative of Standard Deviation of Spectral Flux
Compactness
Derivative of Compactness
Running Mean of Compactness
Standard Deviation of Compactness
Add Recordings
Delete Recordings
Save Features For Each Window
Save For Overall Recordings
Feature Values Save Path:
Softpedia.xml
Window Size (samples):
512
Feature Definitions Save Path:
Softpedia_definitions_1.xml
Window Overlap (fraction):
0.0
Alter Aggregators
Extract Features

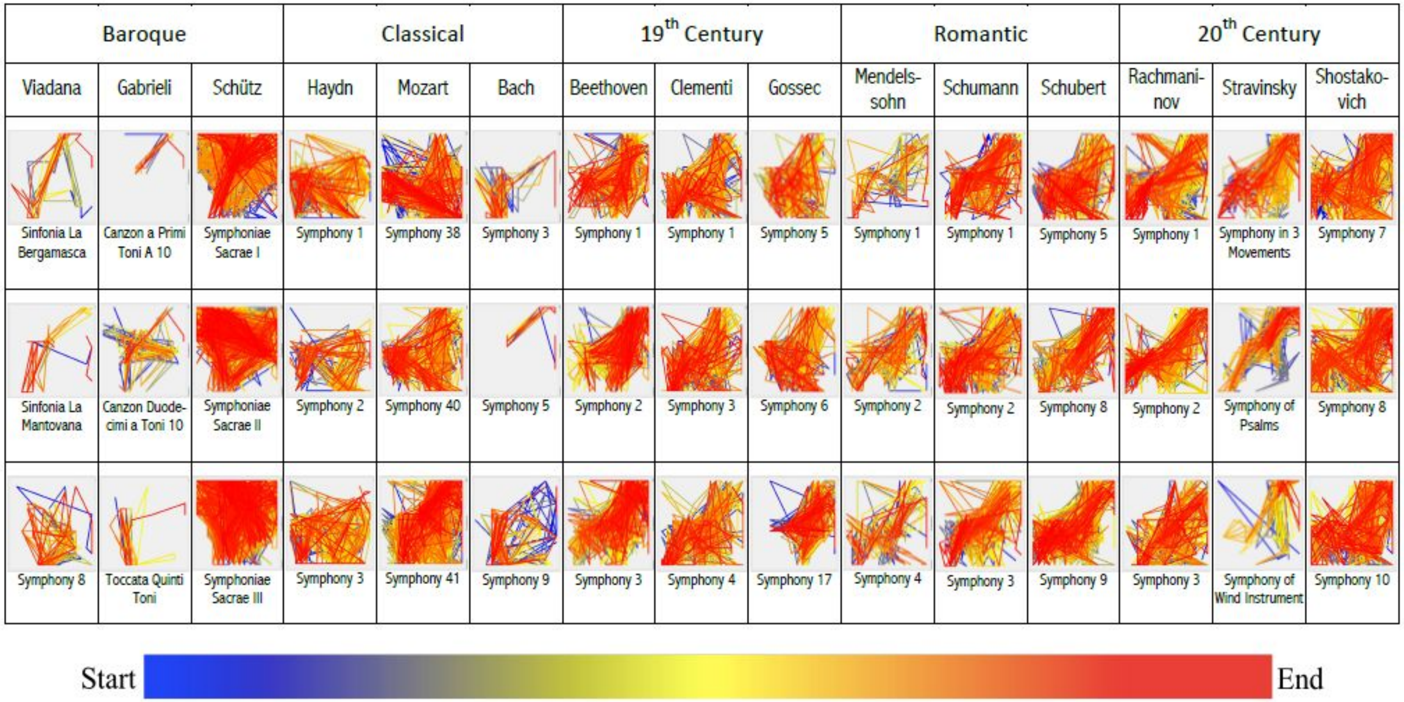
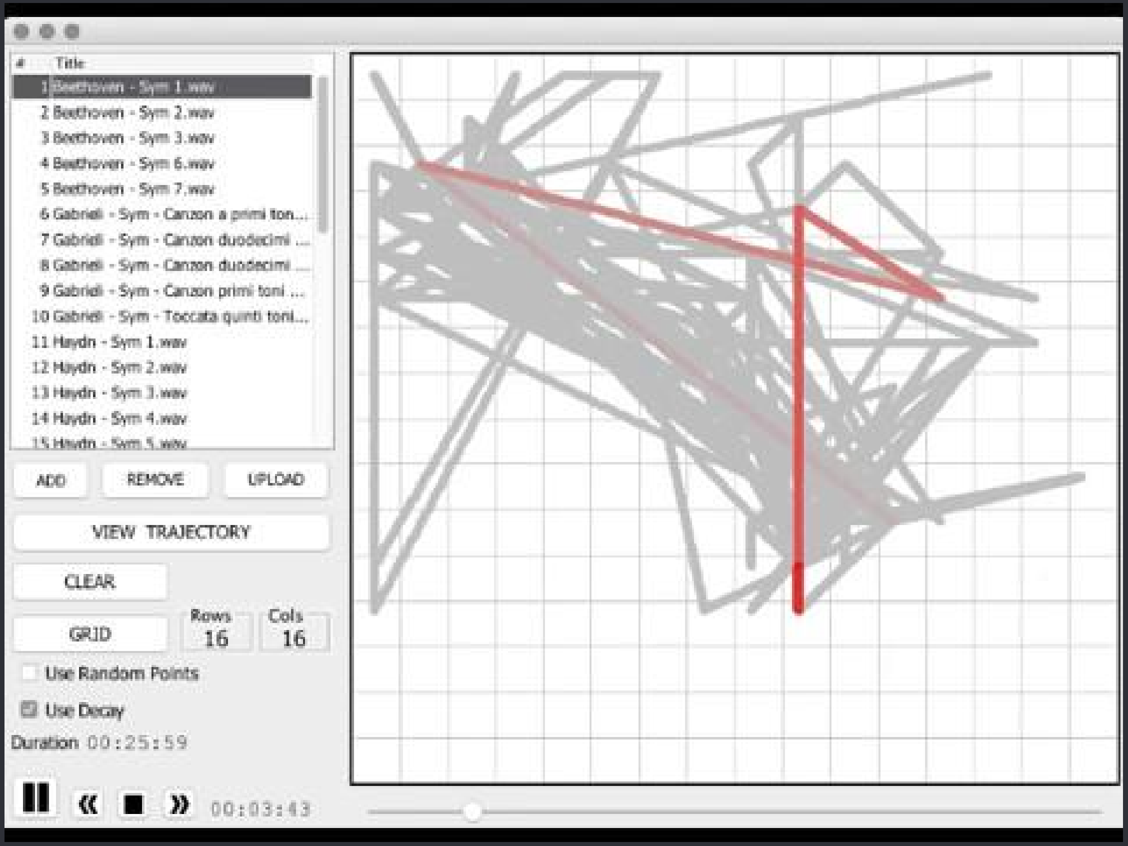


Fig. 2. SOMphony Trajectories and Color Spectrum to designate time in the SOMphony Map

http://tinyurl.com/SOMphony

SOMphony map using k=21



Title
1 Beethoven - Sym 1.wav
2 Beethoven - Sym 2.wav
3 Beethoven - Sym 3.wav
4 Beethoven - Sym 6.wav
5 Beethoven - Sym 7.wav
6 Gabrieli - Sym - Canzon a primi ton...
7 Gabrieli - Sym - Canzon duodecimi ...
8 Gabrieli - Sym - Canzon duodecimi ...
9 Gabrieli - Sym - Canzon primi toni ...
10 Gabrieli - Sym - Toccata quinti toni...
11 Haydn - Sym 1.wav
12 Haydn - Sym 2.wav
13 Haydn - Sym 3.wav
14 Haydn - Sym 4.wav
ADD
REMOVE
UPLOAD
VIEW TRAJECTORY
CLEAR
GRID
Rows
16
Cols
16
Use Random Points
Use Decay
Duration 00:25:59
00:03:43

Research Gap

- Using normalized frequency count as a basis for clustering does not consider the notion of time.
- The sequence of music with regard to time is not considered.

General Objective

To develop a 3D visualization model that incorporates time series in comparing symphonies using 3D SOM's

Specific Objective # 1

Objective #1

To include more symphonies to the data set

Scope and Limitations

- Expand the previous data set to have 5 symphonies per composer
- Composers still the same as previous data set
- Quality of music data is disregarded if limited

Specific Objective # 2

Objective #2

To determine optimal features to be used

Scope and Limitations

- Music features that can be extracted from JAudio
- Limit features to top 20 features based on decision tree (top-down)

Specific Objective # 3

Objective #3

To add the in the time series variable

Scope and Limitations

- 0.5 second overlap
- Each SOM will be assigned to a 1 sec segment

Specific Objective # 4

Objective #4

To create a 3D visualization model for the data

Scope and Limitations

- OpenGL for visualization
- Representing each map in a time series

Specific Objective # 5

Objective #5

To have participants listen and annotate the musical pieces for qualitative data

Scope and Limitations

- 50 Participants, with musical inclination over a period of 2 months.
- 5 symphonies deemed by the algorithm to have the highest % of similarity

Specific Objective # 6

Objective #6

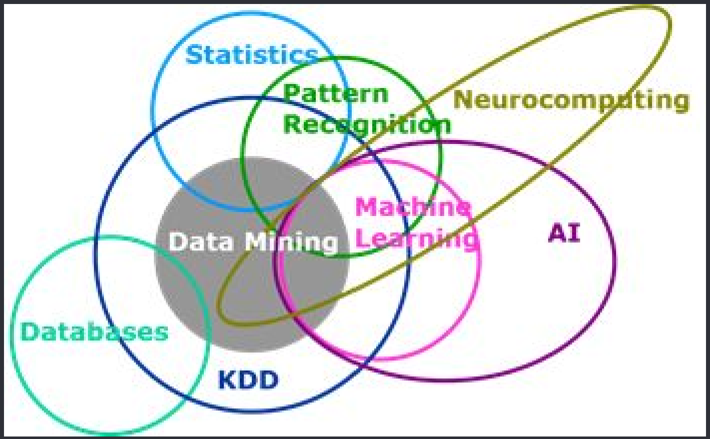
To verify the results of the 3D SOMphony through the results obtained from the human participants

Scope and Limitations

- Results from the participants will be compared to the results of 3D SOMphony
- Only music samples used in the qualitative data

Machine Learning

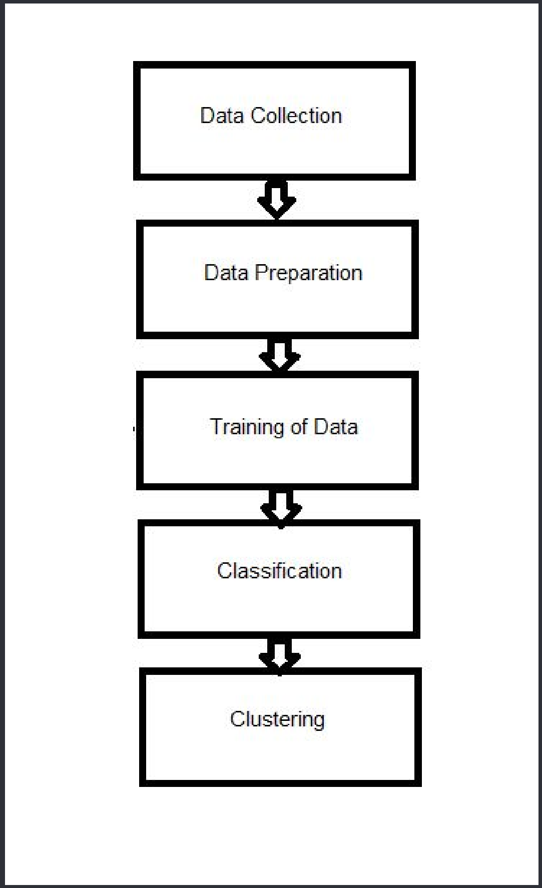
Explore possible application of research to existing fields in machine learning



Research Significance

Methodology and Experiments

Serve as basis and reference for future research related to music feature visualization and analysis



Research Significance

Related Systems

Results of this study can be further used to improve systems such as Automatic Playlist Generation or studies on Music Theory

© Rachel Wells

Automatic Playlist Generation (Xingting Gong & Xu Chen, Stanford University)

Research Significance

Fields Outside Computer Science

The findings in this research may be used in almost any field that is time sensitive such as network traffic.

© Jose Lepez

Research Methodology

Research Activities

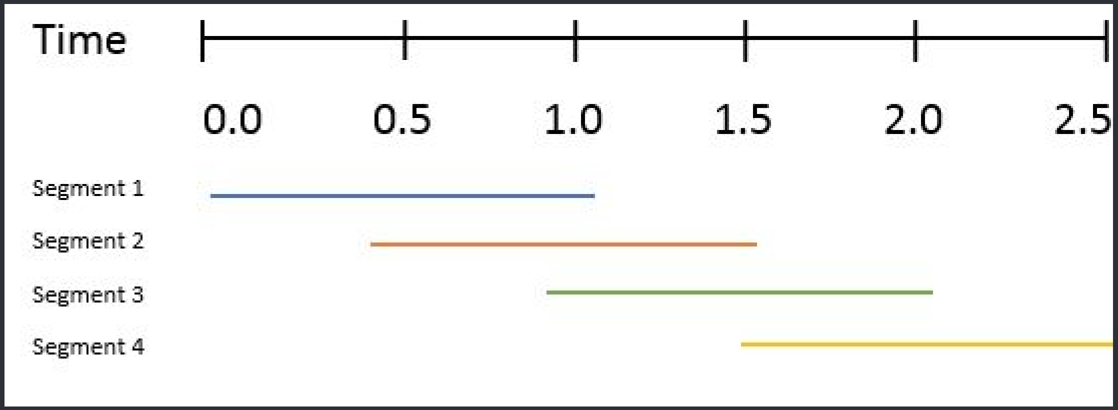
- Concept Formulation and Review of Related Literature
- Data Gathering
- Pre-processing
- Training
- Visualization Development
- Performance Evaluation and Human Evaluation
- Data Analysis
- Documentation

Data Gathering

- Additional 2 symphonies per composer
- Obtained through online or physical means (Youtube, CD’s)
- File type and bitrate are not taken into consideration
- Audio quality is disregarded

Pre-Processing

- Audio files would be converted into wav files in preparation for splitting
- Split audio files into 1 second segments overlapping at 0.5 second using **WaveSplitter**



Pre-Processing

- Segments will undergo feature extraction using jAudio.
- Run RegEx script on the .xml file to extract the unnecessary text in preparation for labeling.
- Convert resulting file to .csv

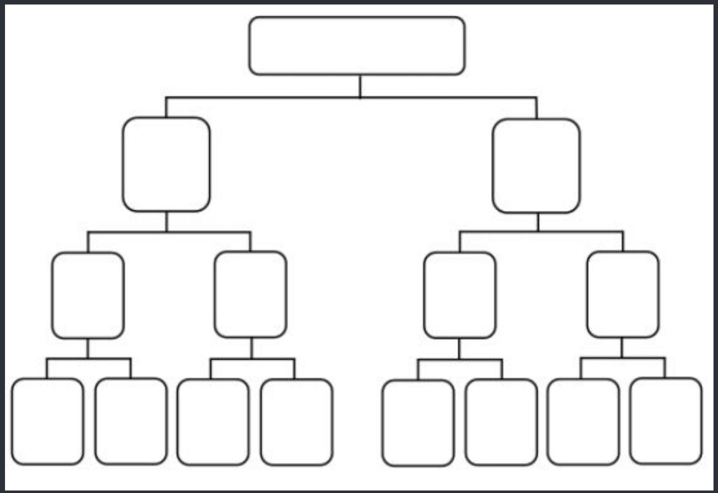
Pre-Processing

- Label the excel file columns
  - Composer (A)
  - Composition (B)
  - Segment Name (C)

	A	B	C
1	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 001.wav
2	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 002.wav
3	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 003.wav
4	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 004.wav
5	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 005.wav
6	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 006.wav
7	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 007.wav
8	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 008.wav
9	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 009.wav
10	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 010.wav
11	Beethoven	Beethoven Sym No. 1	[Beethoven] Symphony No. 1 - 1. Adagio molto - Allegro con brio 011.wav

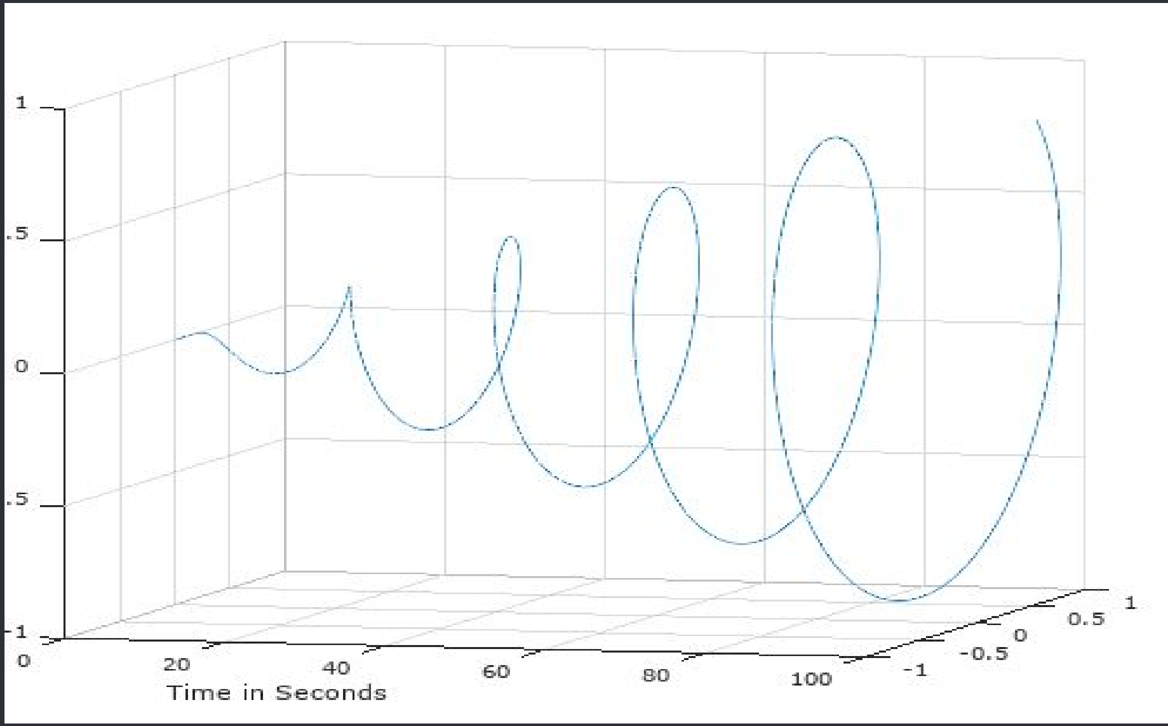
Feature Selection

- Initial feature selection results to at most 600 features.
- Trim down to 20 features using **decision tree learning**.
- First 20 nodes (top-down) would be selected as the top features.



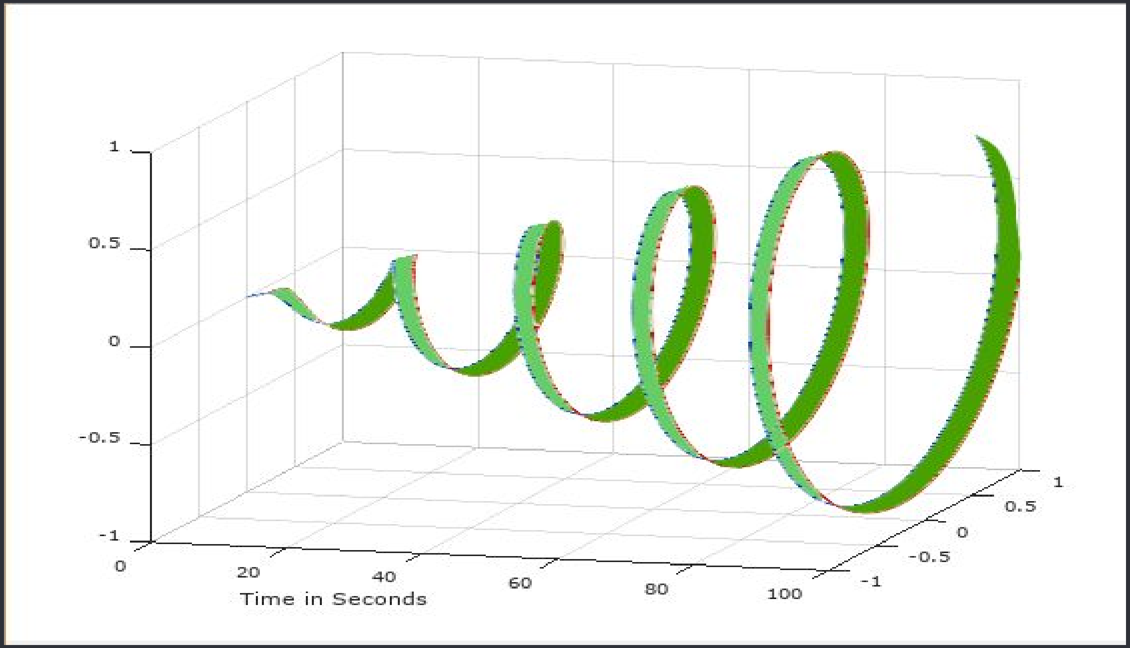
Visualization

- OpenGL
- Euclidean Distance



Performance Evaluation

- 50 human participants within 2 months
- Knowledgeable in music
- To annotate marked regions in the player.
- Also given the freedom to annotate unmarked regions.


1
0.5
0
-0.5
-1
0
20
40
60
80
100
Time in Seconds
1
0.5
0
-0.5
-1


ChicagoSymphony
5 days
Stravinsky Scenes De Ballet
16:43

Calendar of Activities for 2017

Activities	JUN	JUL	AUG	SEPT	OCT	NOV	DEC
1	♫♫♫	♫♫♫	♫				
2			♫	♫♫♫	♫♫		
3				♫♫	♫♫	♫♫♫♫	♫♫
4						♫♫	♫♫
5				♫♫	♫♫	♫♫	♫
6							
7	♫♫	♫♫♫	♫♫	♫♫♫	♫♫♫♫	♫♫♫♫	♫♫

1. Concept formulation and RRL
2. Data gathering
3. Pre-processing
4. Feature Selection
5. Visualization Development
6. Performance Evaluation and Human Evaluation
7. Documentation

Legend: ♫ - 1 week (10 hours)

Calendar of Activities for 2018

Activities	JAN	FEB	MAR	APR	MAY	JUN	JUL
1							
2							
3							
4	♫♫♫						
5	♫♫♫	♫♫♫♫					
6			♫♫♫♫	♫♫♫♫	♫♫♫	♫♫♫	
7	♫♫♫	♫♫♫♫	♫♫♫♫	♫♫♫♫	♫♫♫♫	♫♫♫♫	♫♫

Legend: ♫ - 1 week (10 hours)

1. Concept formulation and RRL
2. Data gathering
3. Pre-processing
4. Feature Selection
5. Visualization Development
6. Performance Evaluation and Human Evaluation
7. Documentation

Summary of Proposal

- Using normalized frequency count as a basis for clustering does not consider the notion of time.
- To develop 3D visualization method for SOMs

End of Presentation

Thank you for listening!

Self-Organizing Maps (SOM)

- Input space is represented into a 2D
- Used to encode the musical trajectory of the different symphonies for visual analysis

K-means Clustering

Used to partition similar nodes from the SOM Map

- Nodes in close proximity get grouped into a cluster
- Clusters represent similarly sounding segments of music
- Each music segment will have a Best Matching Unit that assigns where it belongs in a cluster

Time Series

- Serial data that includes equally divided points in time order

Visualization

- 2D to 3D using OpenGL as visualization