

# Music Information Retrieval

## Music Feature Extraction

<http://www.ifs.tuwien.ac.at/mir>



**Alexander Schindler**

Institute of Software Technology and Interactive Systems  
Vienna University of Technology  
[schindler@ifs.tuwien.ac.at](mailto:schindler@ifs.tuwien.ac.at)



## What is Music IR?

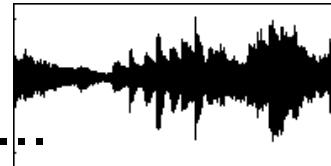
- Searching for Music
  - Searching for music on the Web
  - Query by Humming
  - Similarity Retrieval
  - Identity detection (fingerprinting)
- Extraction of information from music
  - plenty of other tasks!

# What is Music?

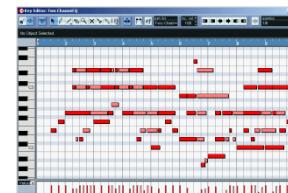


## ■ Music

Audio: wav, au, mp3, ...



Symbolic: MIDI, mod, ...



Scores: Scan, MusicXML



## ■ Text

- Song lyrics
- Artist Biographies
- Websites:  
Fanpages, Blogs,  
Album Reviews,  
Genre descriptions

## ■ Community data

- Market basket
- Tags
- Social Networks
- Spotify
- Last.fm

## ■ Video/Images

- Album covers
- Music videos

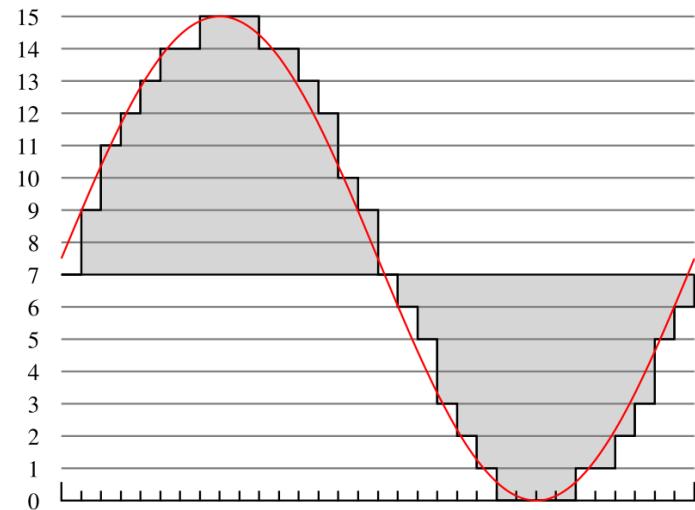
2.

## Feature Extraction from Music

# Too much Audio Data



- Digital Audio
  - Sampling Rate: 44,100 Hz
  - 16-bit resolution for each channel
    - 2 channels for stereo
  - 88,200 Integers per second



# Excercise: Find Documents Containing the Word „Music“



## Document 1:

*“Most of these issues stem from the commercial interest in **music** by record labels, and therefore imposed rigid copyright issues, that prevent researchers from sharing their **music** collections with others. Subsequently, only a limited number of data sets has risen to a pseudo benchmark level, i.e. where most of the researchers in the field have access to the same collection.”*

## Document 2:

*“The Echonest Analyzer [5] is a **music** audio analysis tool available as a free Web service accessible over the Echonest API and as a commercially distributed standalone command line tool. The Analyzer implements an onset detector which is used for segmentation.”*

## Document 3:

*“The Million Song Dataset (MSD), a collection of one million **music** pieces, enables a new era of research of **Music** Information Retrieval methods for large-scale applications. It comes as a collection of meta-data such as the song names, artists and albums, together with a set of features extracted with the The Echo Nest services, such as loudness, tempo, and MFCC-like features.”*

# Excercise: Find Songs with Strings



## Song 1:

83, 58, 11, 11, 9, 60, 96, 25, 39, 42, 87, 90, 12, 26, 99, 69, 10, 56, 64, 41, 47, 61, 6, 40, 94, 23, 43, 52, 31, 77, 32, 57, 40, 89, 91, 28, 38, 96, 3, 90, 43, 18, 25, 16, 79, 97, 83, 64, 46, 70, 63, 34, 38, 39, 7, 66, 89, 95, 9, 47, 11, 59, 9, 17, 46, 92, 27, 58, 87, 46, 39, 100, 10, 2, 5, 53, 73, 56, 43, 46, 47, 67, 2, 60, 9, 23, 43, 21, 98, 34, 29, 62, 26, 72, 38, 98, 55, 96, 11, 49, 83, 58, 11, 11, 9, 60, 96, 25, 39, 42, 87, 90, 12, 26, 99, 69, 10, 56, 64, 41, 47, 61, 6, 40, 94, 23, 43, 52, 31, 77, 32, 57, 40, 89, 91, 28, 38, 96, 3, 90, 43, 18, 25, 16, 79, 97, 83, 64, 46, 70, 63, 34, 38, 39, 7, 66, 89, 95, 9, 47, 11, 59, 9, 17, 46, 92, 27, 58, 87, 46, 39, 100, 10, 2, 5, 53, 73, 56, 43, 46, 47, 67, 2, 60, 9, 23, 43, 21, 98, 34, 29, 62, 26, 72, 38, 98, 55, 96, 11, 49, 83, 58, 11, 11, 9, 60, 96, 25, 39, 42, 87, 90, 12, 26, 99, 69, 10, 56, 64, 41, 47, 61, 6, 40, 94, 23, 43, 52, 31, 77, 32, 57, 40, 89, 91, 28, 38, 96, 3, 90, 43, 18, 25, 16, 79, 97, 83, 64, 46, 70, 63, 34, 38, 39, 7

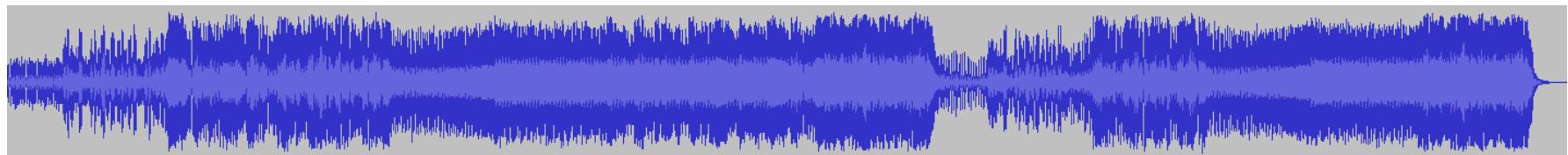
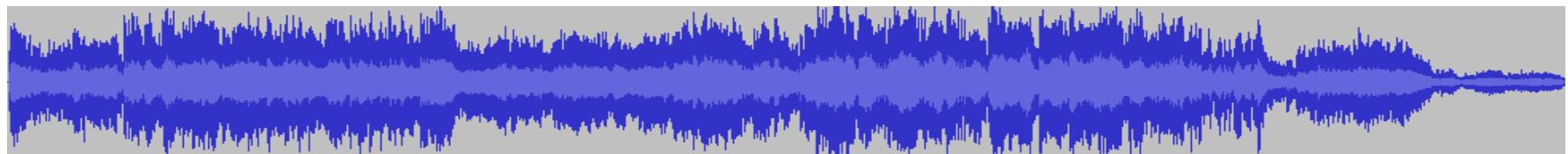
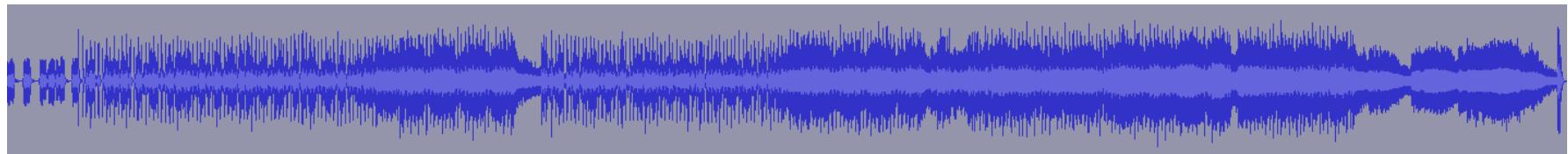
## Song 2:

55, 96, 11, 49, 83, 58, 11, 11, 9, 60, 96, 25, 39, 42, 87, 90, 12, 26, 99, 69, 10, 56, 64, 41, 47, 61, 6, 40, 94, 23, 43, 52, 31, 77, 32, 57, 40, 89, 91, 28, 38, 96, 3, 90, 43, 18, 25, 16, 79, 97, 83, 64, 46, 70, 63, 34, 38, 39, 7, 66, 89, 95, 9, 47, 11, 59, 9, 17, 46, 92, 27, 58, 87, 46, 39, 100, 10, 2, 5, 53, 73, 56, 43, 46, 47, 67, 2, 60, 9, 23, 43, 21, 98, 34, 29, 62, 26, 72, 38, 98, 55, 96, 11, 49, 83, 58, 11, 11, 9, 60, 96, 25, 39, 42, 87, 90, 12, 26, 99, 69, 10, 56, 64, 41, 47, 61, 6, 40, 94, 23, 43, 52, 31, 77, 32, 57, 40, 89, 91, 28, 38, 96, 3, 90, 43, 18, 25, 16, 79, 97, 83, 64, 46, 70, 63, 34, 38, 39, 7

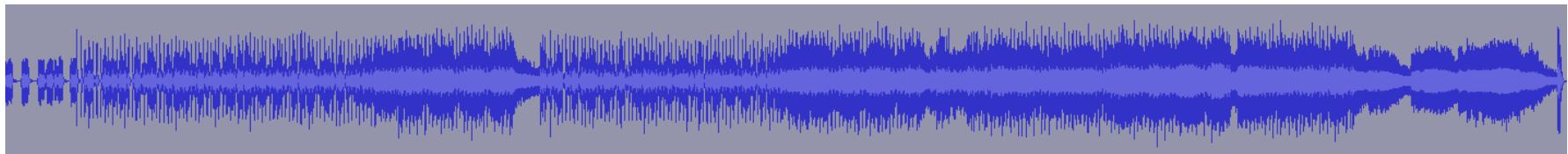
## Song 3:

66, 89, 95, 9, 47, 11, 59, 9, 17, 46, 92, 27, 58, 87, 46, 39, 100, 10, 2, 5, 53, 73, 56, 43, 46, 47, 67, 2, 60, 9, 23, 43, 21, 98, 34, 29, 62, 26, 72, 38, 98, 55, 96, 11, 49, 83, 58, 11, 11, 9, 60, 96, 25, 39, 42, 87, 90, 12, 26, 99, 69, 10, 56, 64, 41, 47, 61, 6, 40, 94, 23, 43, 52, 31, 77, 32, 57, 40, 89, 91, 28, 38, 96, 3, 90, 43, 18, 25, 16, 79, 97, 83, 64, 46, 70, 63, 34, 38, 39, 7, 66, 89, 95, 9, 47, 11, 59, 9, 17, 46, 92, 27, 58, 87, 46, 39, 100, 10, 2, 5, 53, 73, 56, 43, 46, 47, 67, 2

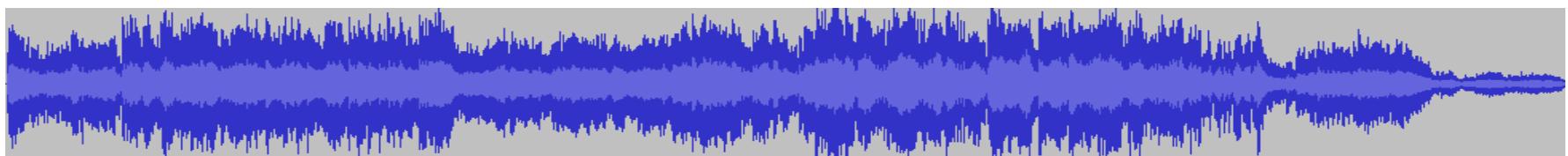
# Excercise: Same Genre?



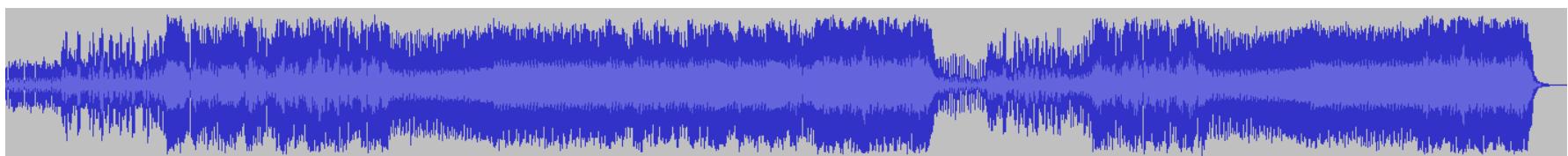
# Excercise: Identify Songs



AC-DC – Highway to Hell



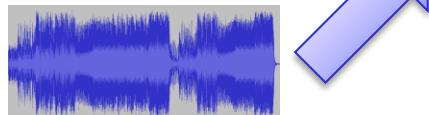
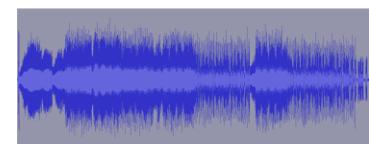
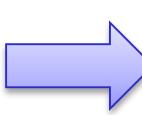
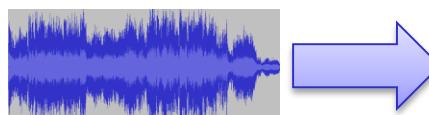
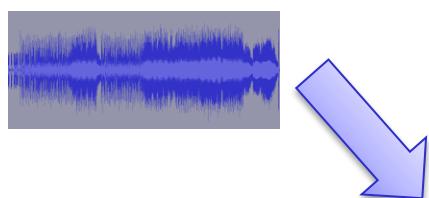
John Williams – Star Wars Main Theme



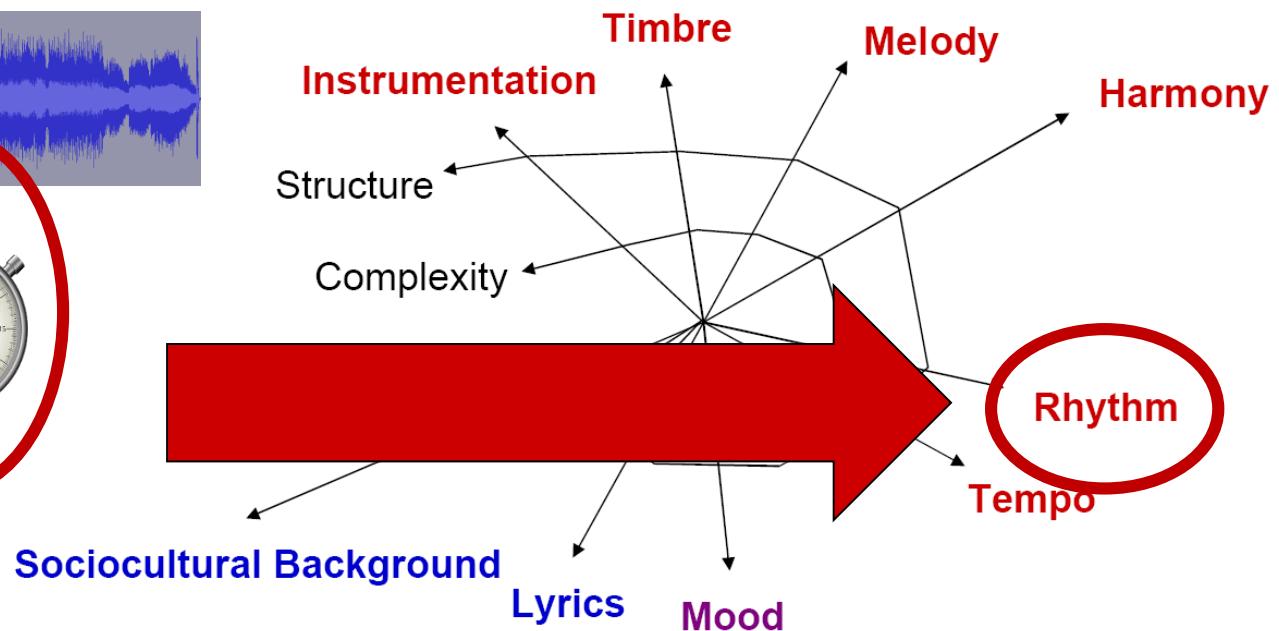
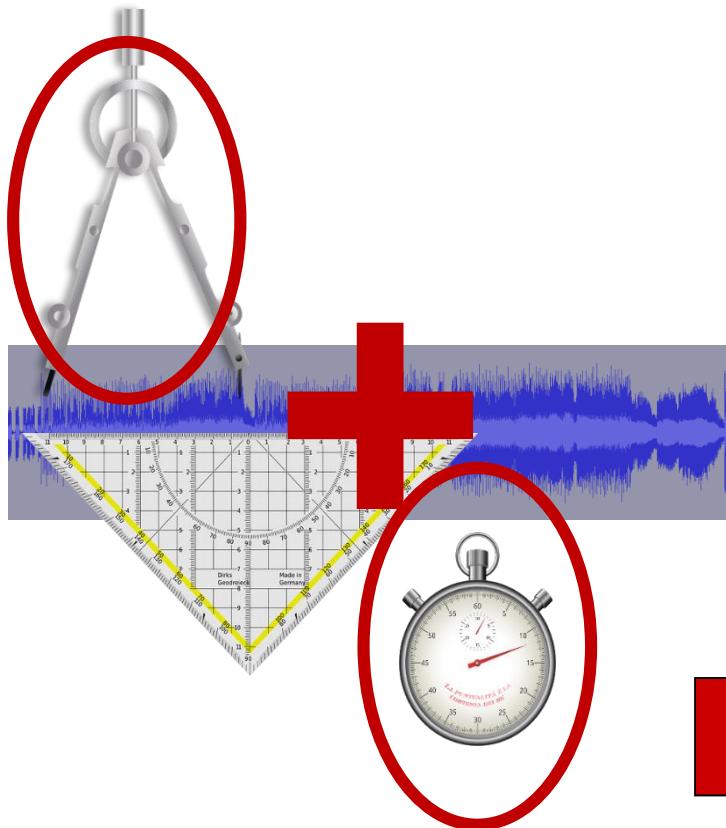
Rihanna feat. Calvin Harris – We Found Love

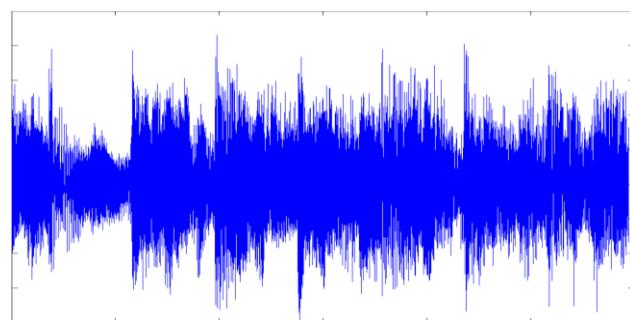
- Reduce audio data by extracting information about:
  - Pitch
  - Timbre
  - Rhythm
  - etc.
- → extract „audio descriptors“

# Problem: Source Separation

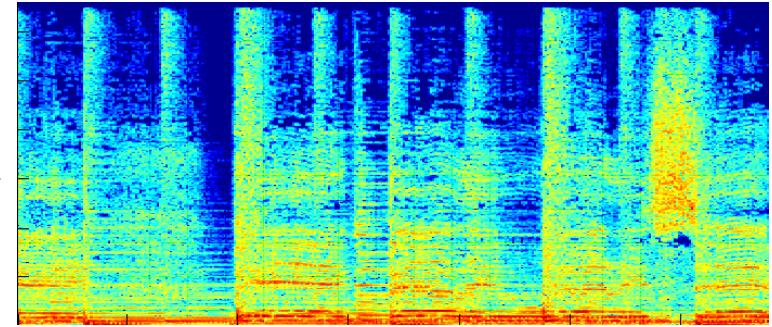


# Audio Feature Extraction





Time Domain  
("Wave Form")



Frequency Domain  
("Spectrum")

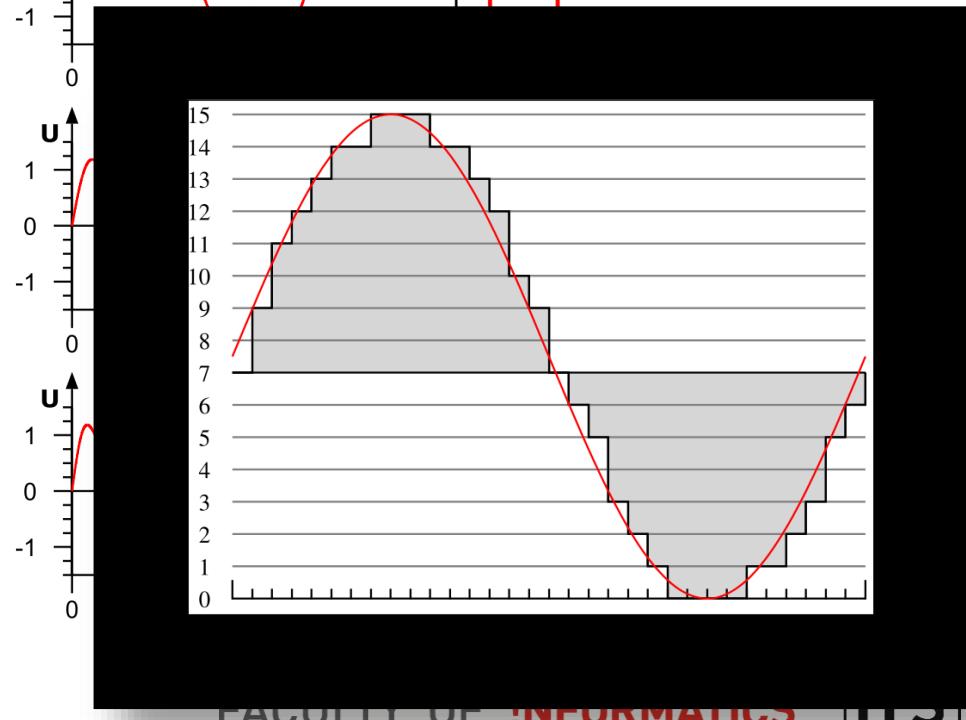
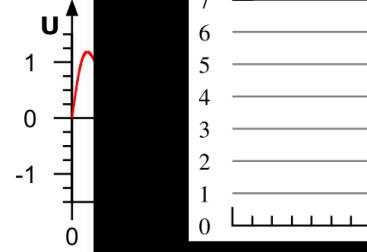
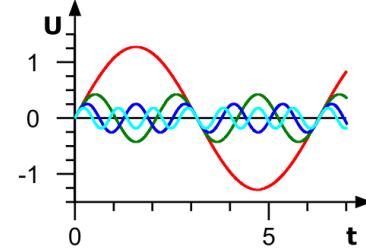
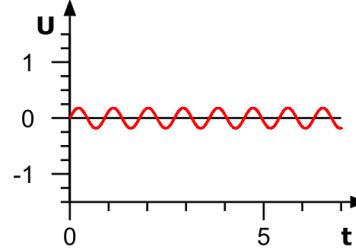
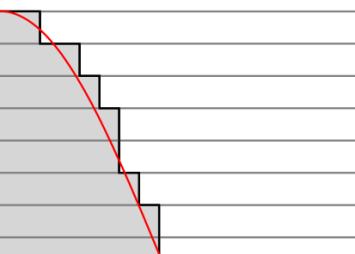
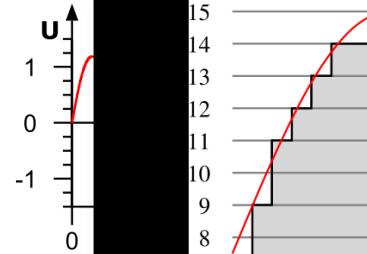
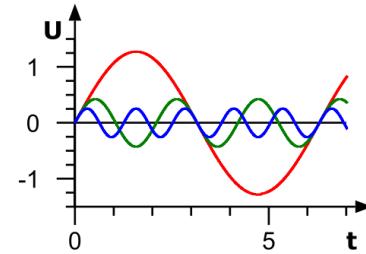
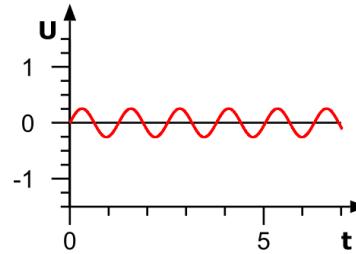
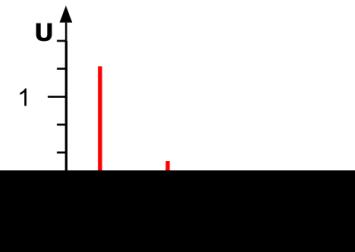
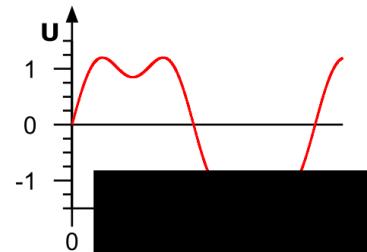
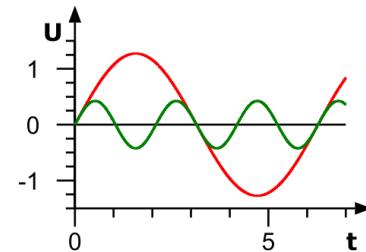
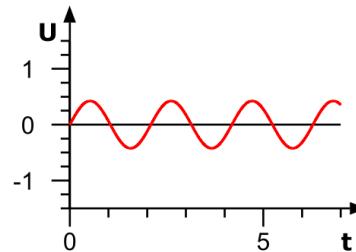
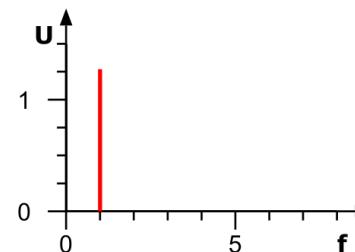
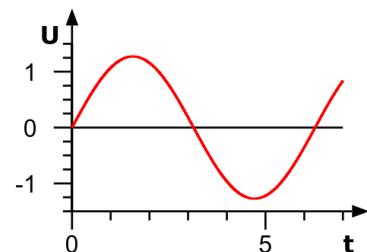
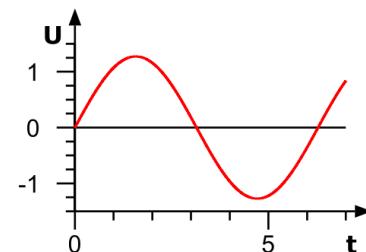
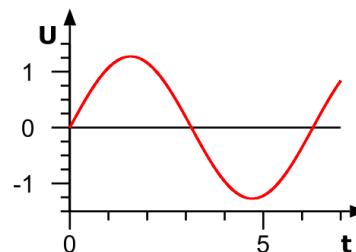
## Time-Frequency Transformation

Fourier Transform (FFT)

Discrete Cosine Transform (DCT)

Wavelet Transform

# Fourier Transform

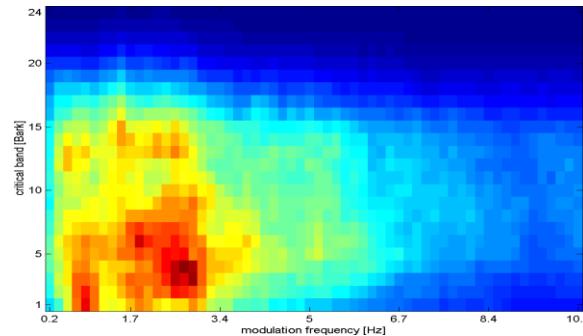


# Audio/Music Feature Extraction by example...

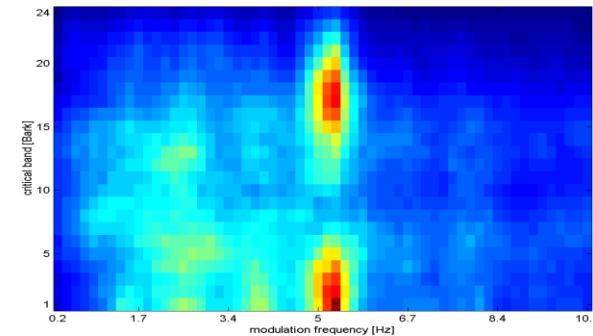
# Rhythm Pattern (RP)



- fluctuations on critical frequency bands  
(a.k.a. Fluctuation Pattern)
- covers rhythm in the broad sense

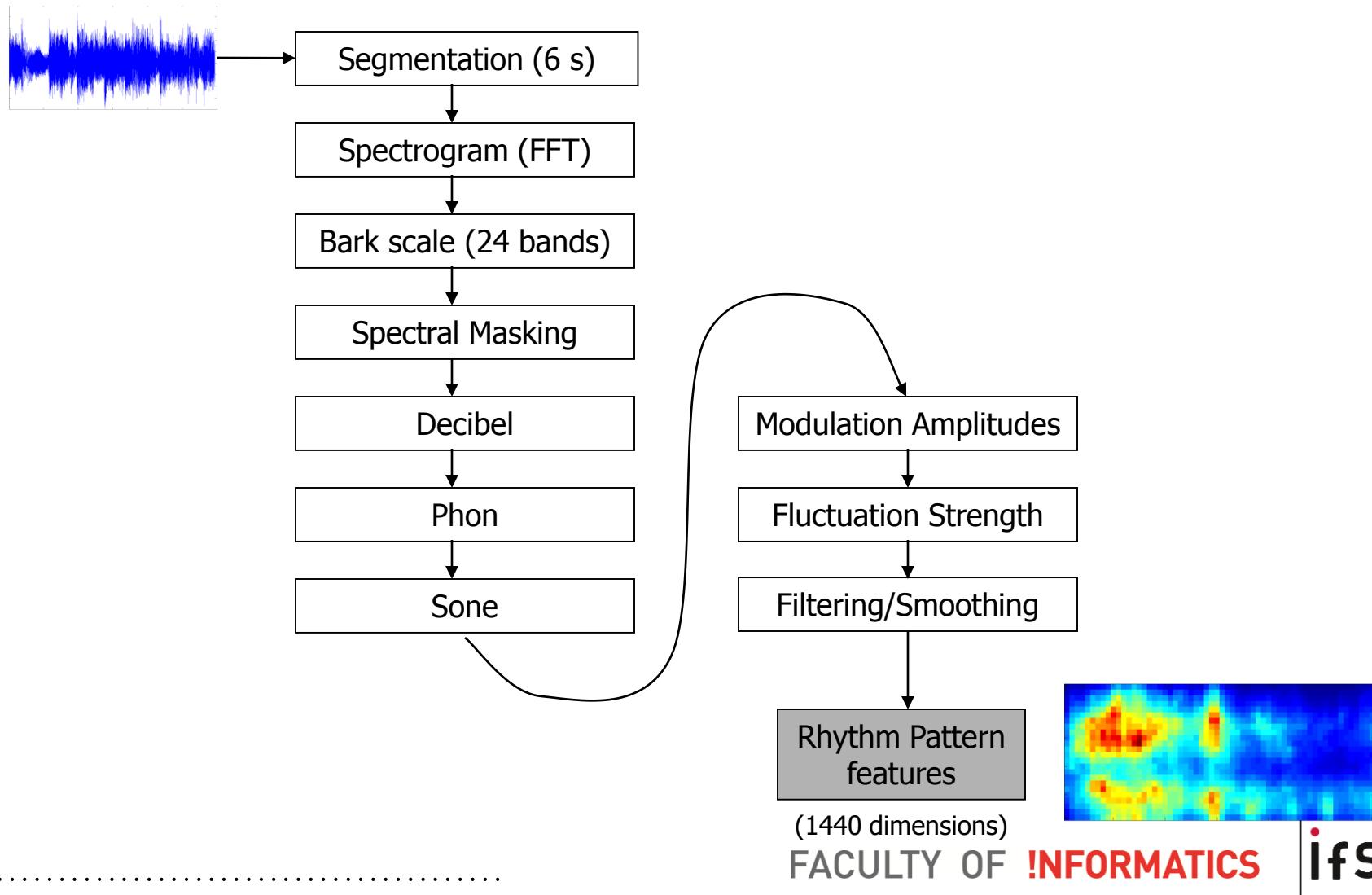


Classical

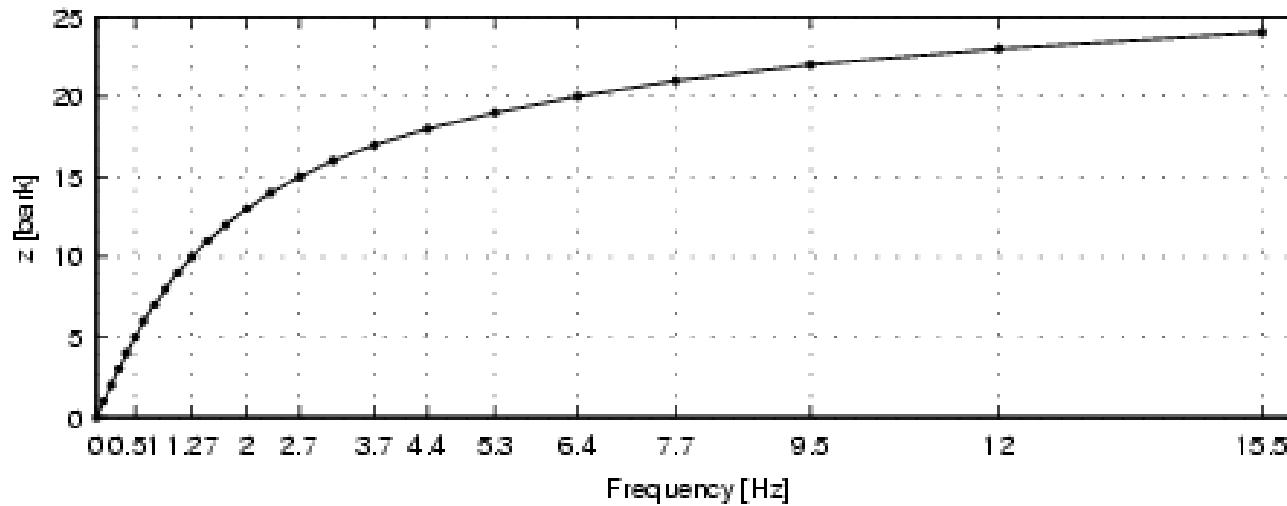


Rock

# Rhythm Pattern (RP)



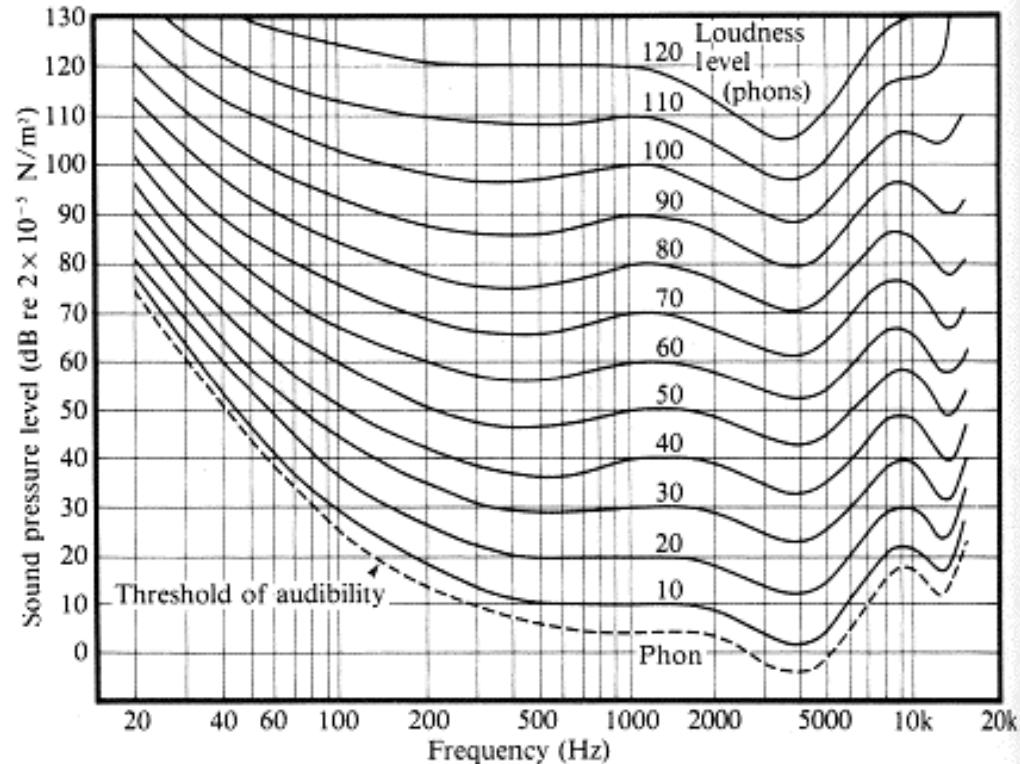
- psychoacoustical scale (related to Mel scale)
- 24 „critical bands“ of hearing (non-linear)
- proposed by Eberhard Zwicker in 1961



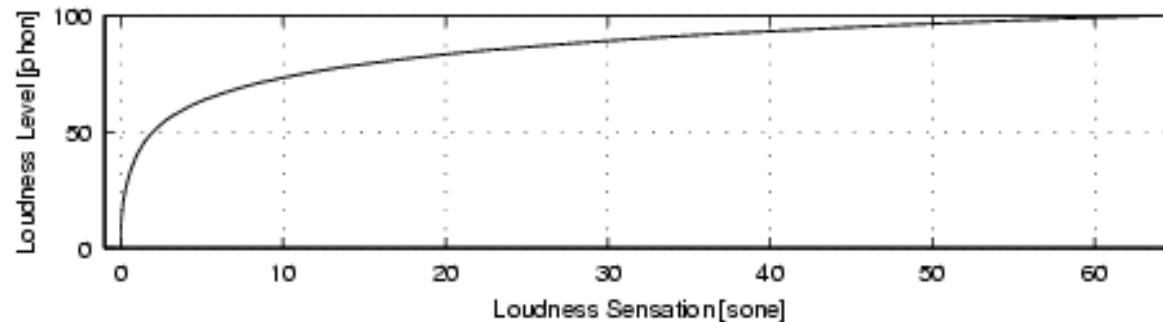
# Equal loudness curves (Phon)



- Relationship between sound pressure level in decibel and hearing sensation is not linear
- Perceived loudness depends on frequency of the tone
- equal loudness contours for 3, 20, 40, 60, 80, 100 phon



on-line test: <http://www.phys.unsw.edu.au/jw/hearing.html>



Sone	1	2	4	8	16	32	64
Phon	40	50	60	70	80	90	100

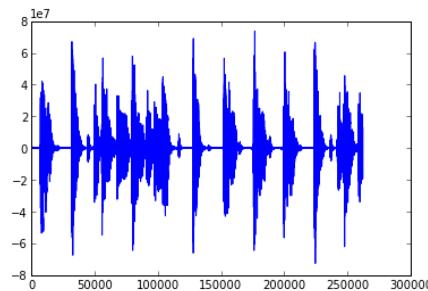
- Perceived loudness measured in Phon does not increase linearly
- Transformation into Sone
- Up to 40 phon slow increase in perceived loudness, then drastic increase
- Higher sensibility for certain loudness differences

# Rhythm Pattern (RP): 2 examples

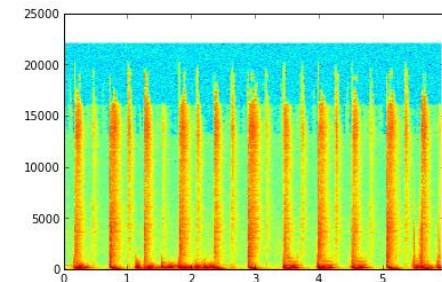


## Queen – Another One Bites The Dust (first 6 seconds)

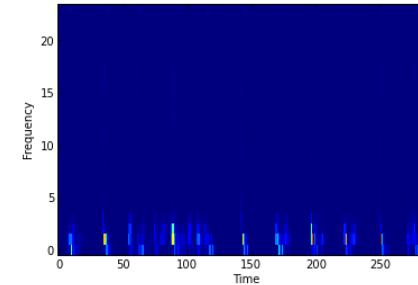
PCM Audio Signal



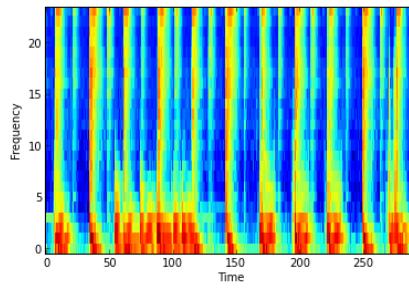
Power Spectrum



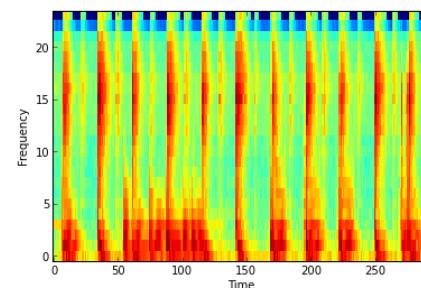
Bark Scale



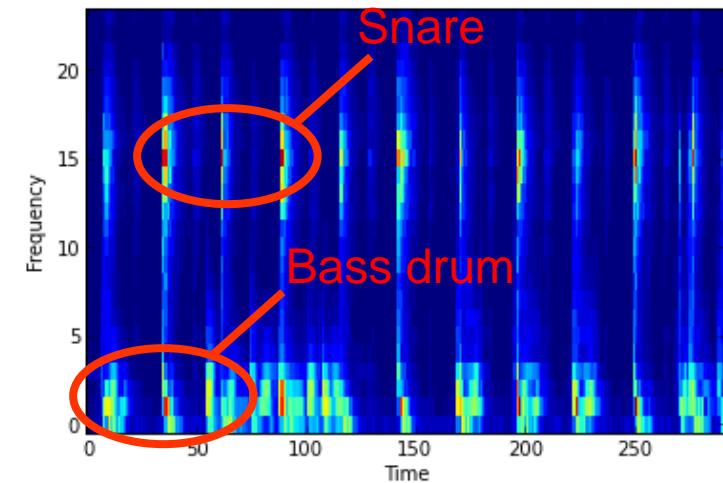
Decibel



Phon



Sone



# Rhythm Pattern (RP): 2 examples



Classical

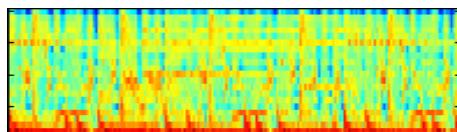
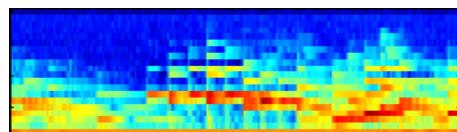


Metal

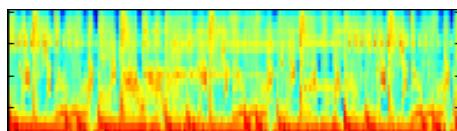
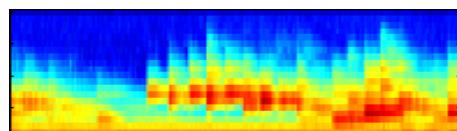


**PCM Audio Signal**

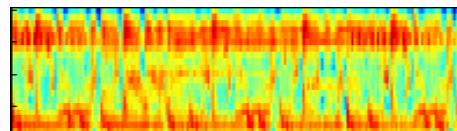
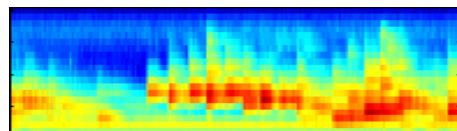
Power Spectrum



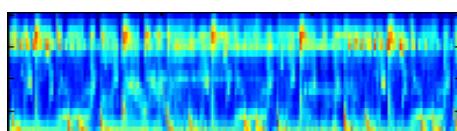
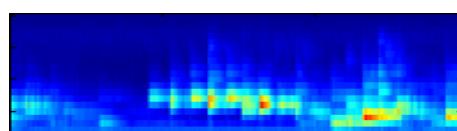
Frequency Bands



Masking Effects



Phon



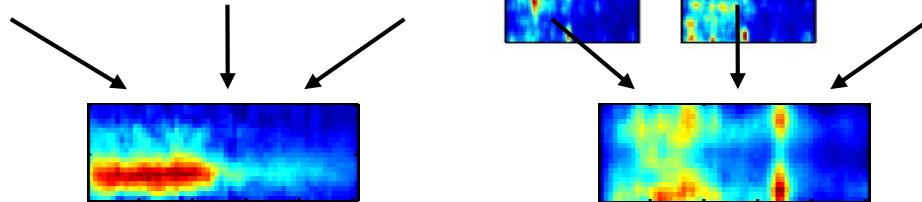
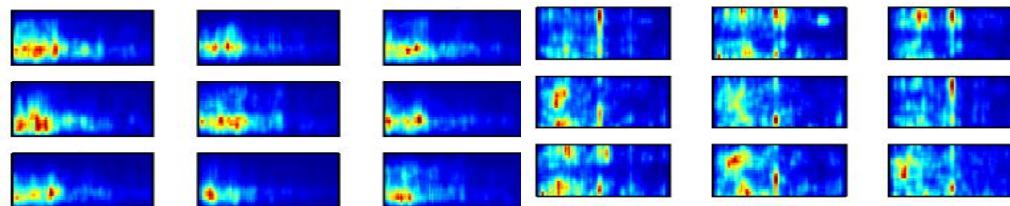
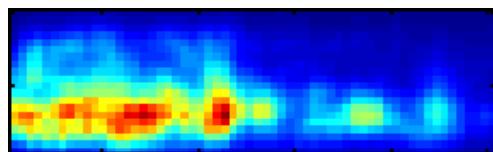
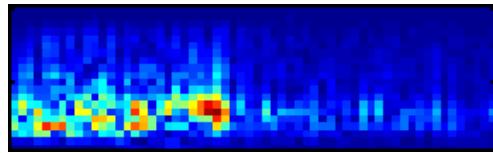
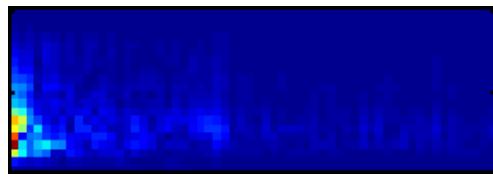
Sone

FACULTY OF **INFORMATICS**

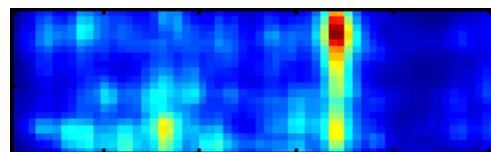
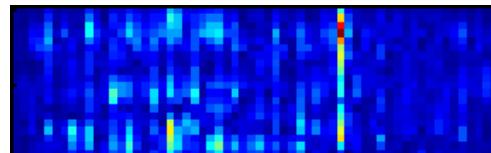
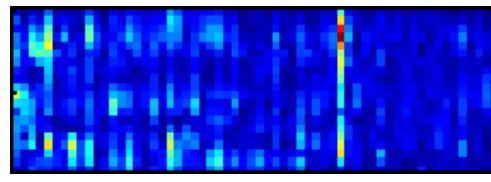
# Rhythm Pattern (RP): 2 examples



Classical



Metal



modulation amplitude  
spectrum (“cepstrum”)

Fluctuation Strength

Filter (Gradient, Gauss)

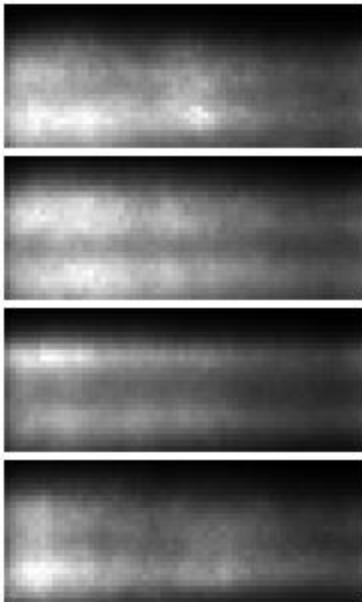
Median

$24 \times 60 =$   
1.440-dim feature vec.

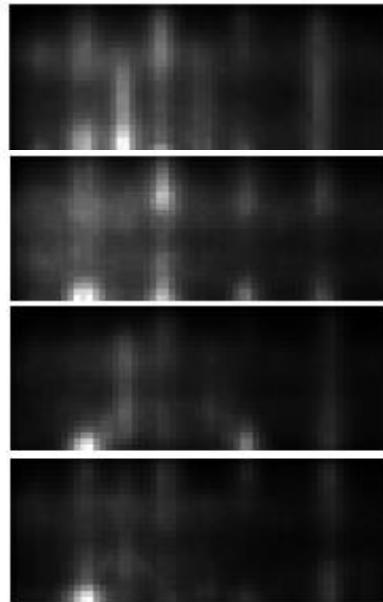
# RP per Genre



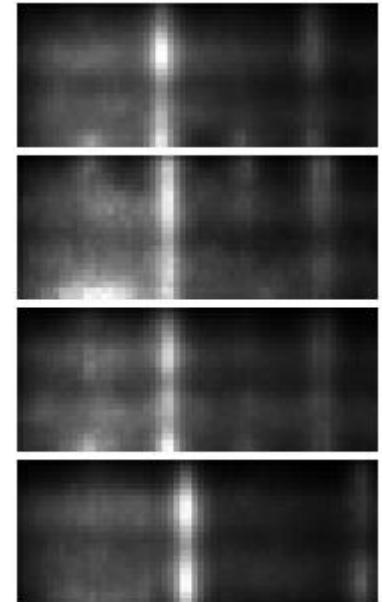
Opera



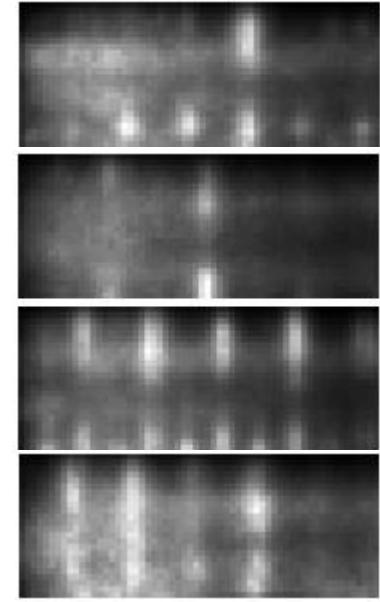
Dance



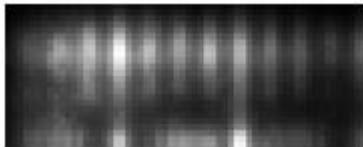
Latin



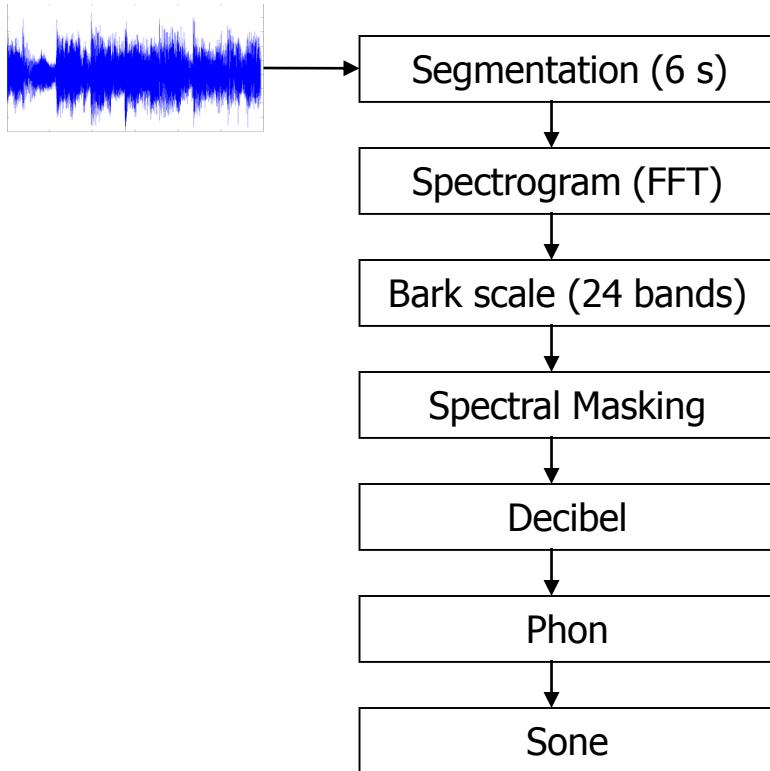
Metal



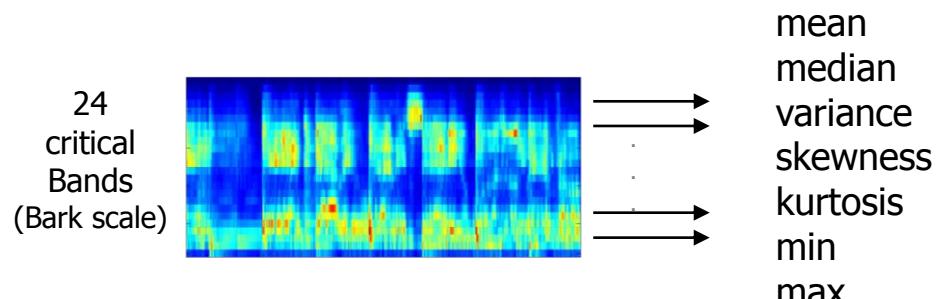
Modulated Synthesizer



# Statistical Spectrum Descriptors

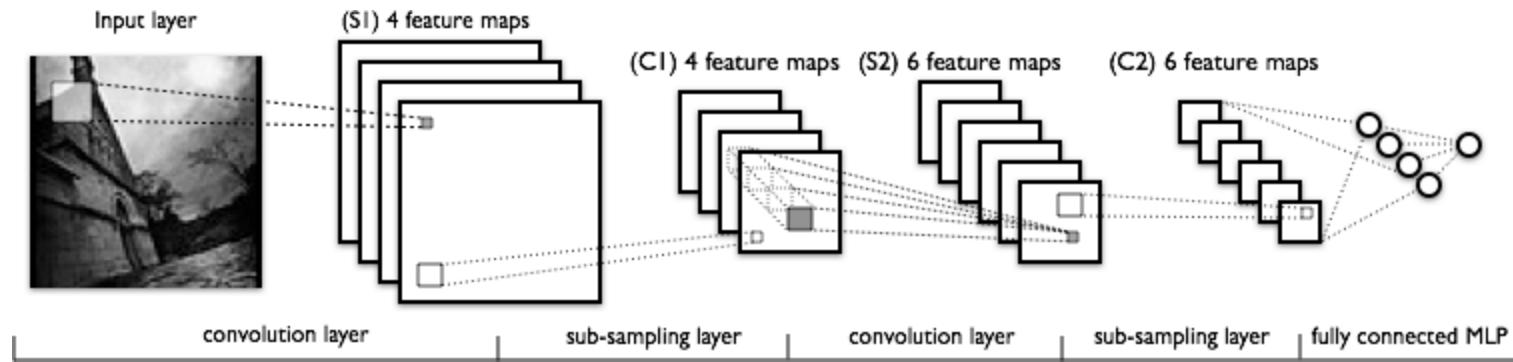


- description of each of the 24 critical bands of the Sonogram by 7 statistical measures
- 168 feature attributes (24x7)



Bark-scale Sonogram (after Sone Step of RP)

# Deep Learning for Music Information Retrieval



<http://deeplearning.net/tutorial/lenet.html>

Combines three types of layers:

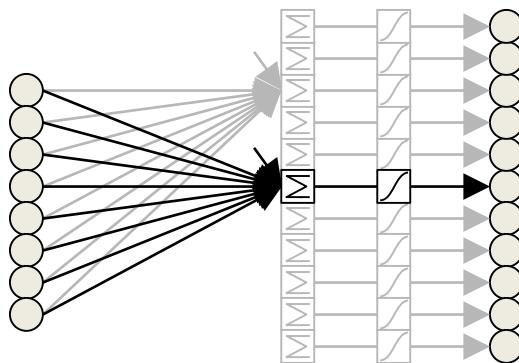
- **Convolutional layer:** performs 2D convolution of 2D input with multiple learned 2D kernels
- **Subsampling layer:** replaces 2D patches by their maximum (“max-pooling”) or average
- **Fully-connected layer:** computes weighted sums of its input with multiple sets of learned coefficients

Applies a nonlinear function after each linear operation (without, a deep network would be linear despite its depth).

# Full vs. Convolutional Layer / Network

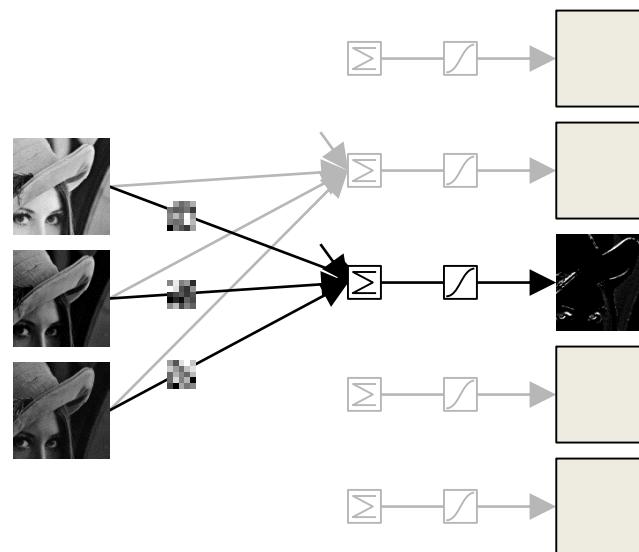
Fully-connected layer:

Each **input** is a **scalar** value,  
each **weight** is a **scalar** value,  
each output is the sum of  
inputs **multiplied** by weights.



Convolutional layer:

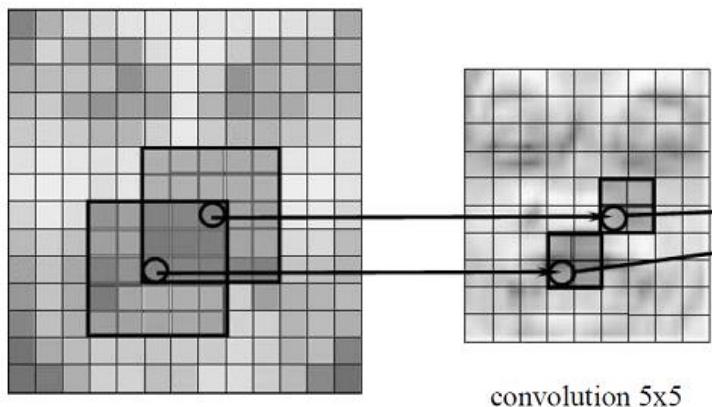
Each **input** is a **tensor** (e.g., 2D),  
each **weight** is a **tensor**,  
each output is the sum of  
inputs **convolved** by weights.



# Motivation for Convolutions



- Apply local filter kernels
- These kernels are the neurons that are learned



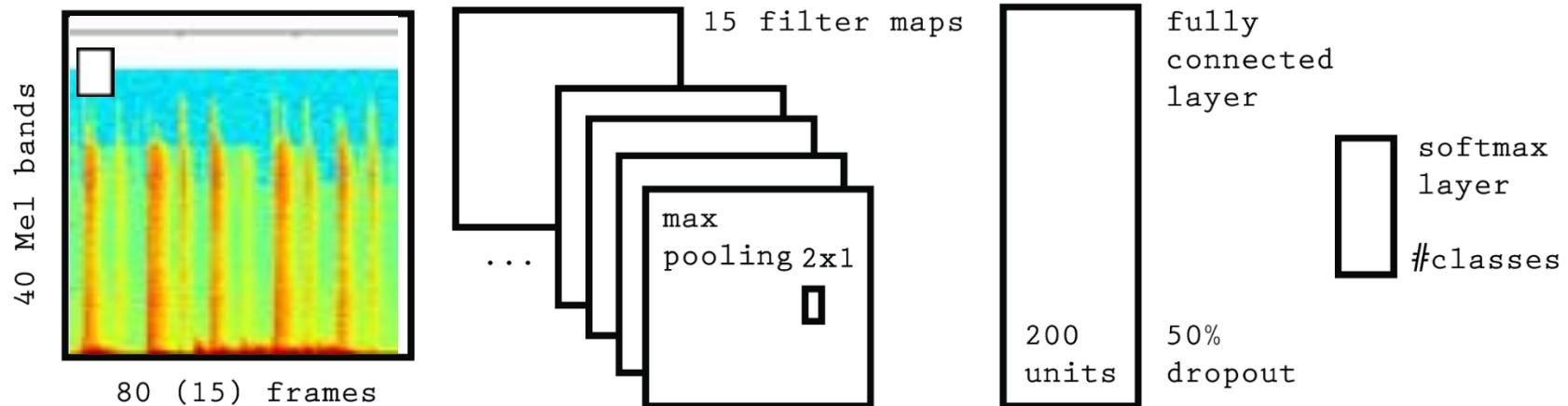
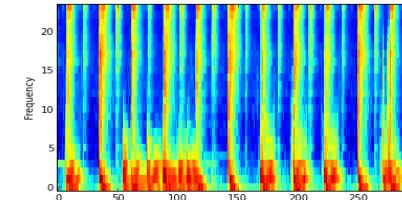
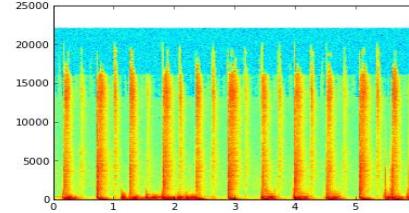
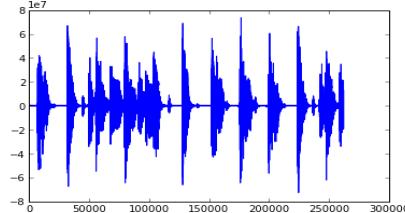
Images: <http://sanghyukchun.github.io/75/>  
[https://en.wikipedia.org/wiki/Kernel\\_\(image\\_processing\)](https://en.wikipedia.org/wiki/Kernel_(image_processing))

Operation	Kernel	Image result
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

# Deep Learning for Music IR



Pre-Processing: Waveform → Spectrogram → 40 Mel bands → Log scale



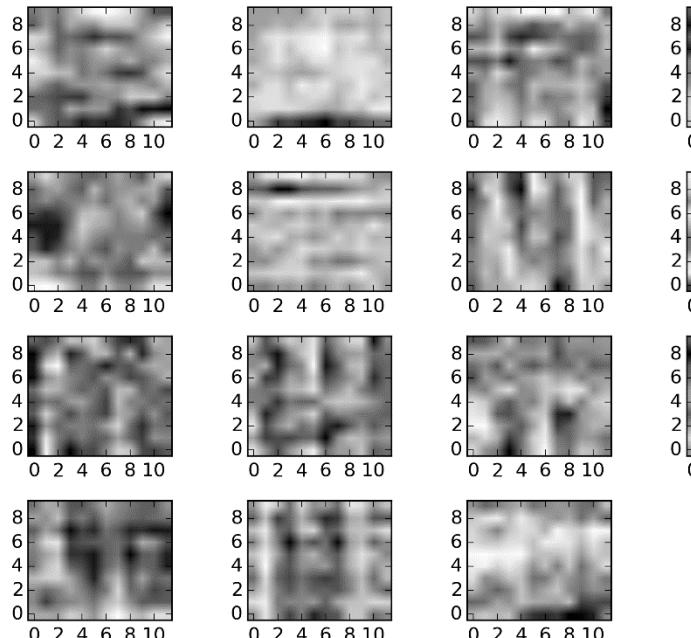
Winning algorithm MIREX 2015 music/speech classification task (99.73%) by Thomas Lidy

# Visualizing CNN Filters

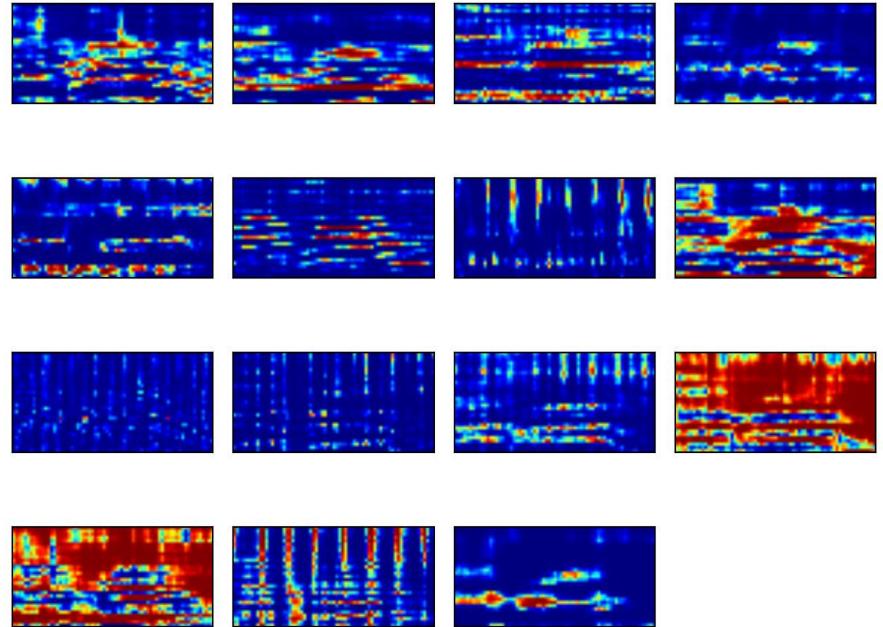
## learned for Music/Speech Classification



Learned Filter Weights

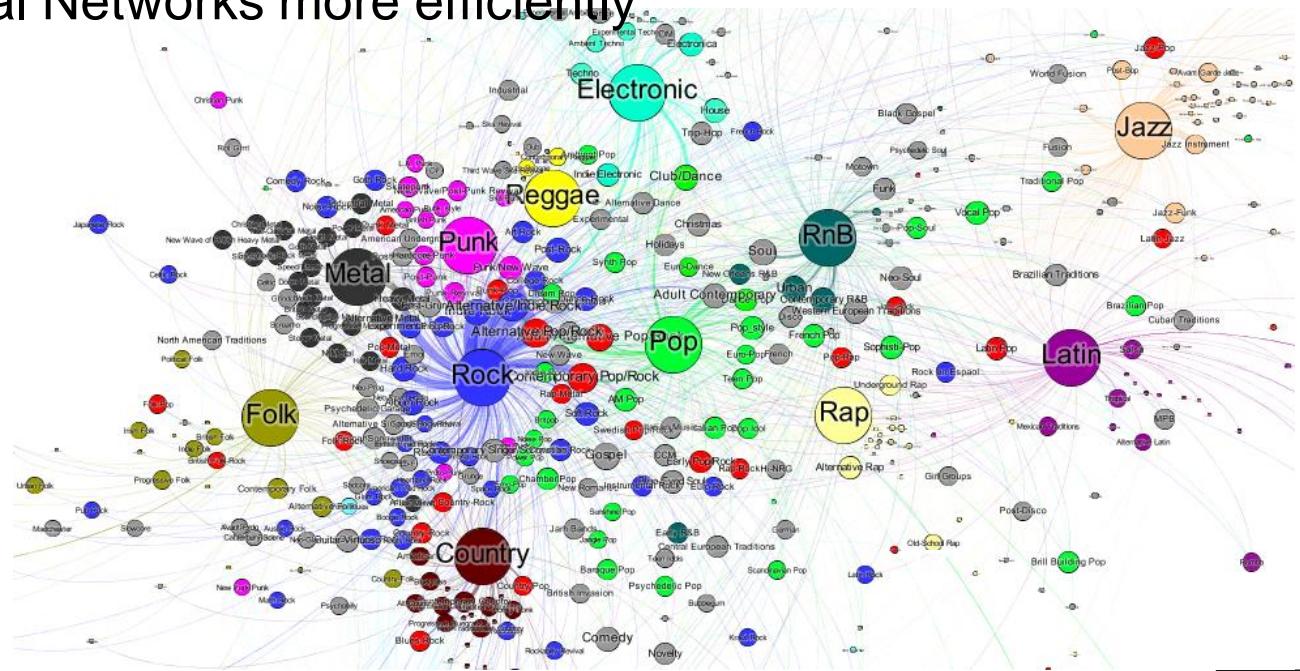


Convolved Spectrograms





- Large scale evaluation
  - Correlation between Genres/Styles/Moods
  - Network Analysis
  - Train Neural Networks more efficiently



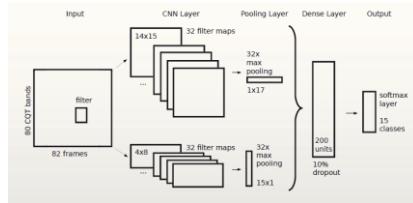
## Audio-Event Detection und Classification

- Urban environments
  - Cars, busses, birds, people talking
- Domestic audio
  - Children playing, turning pages, TV/video games
- Music
  - Beats, segments, verse, chorus, bridge, chord changes



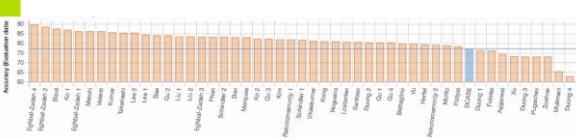
# Urban Sounds Classification

- Detection and Classification of Acoustic Scenes and Events (DCASE)
  - Winning contribution to the Domestic Audio Tagging Task
  - Using a Neural Network based Approach



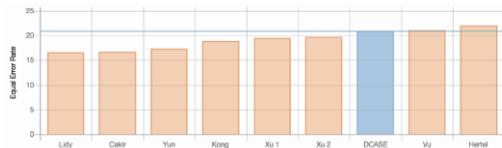
## Task 1 results

- 48 submissions / 34 teams / 113 authors



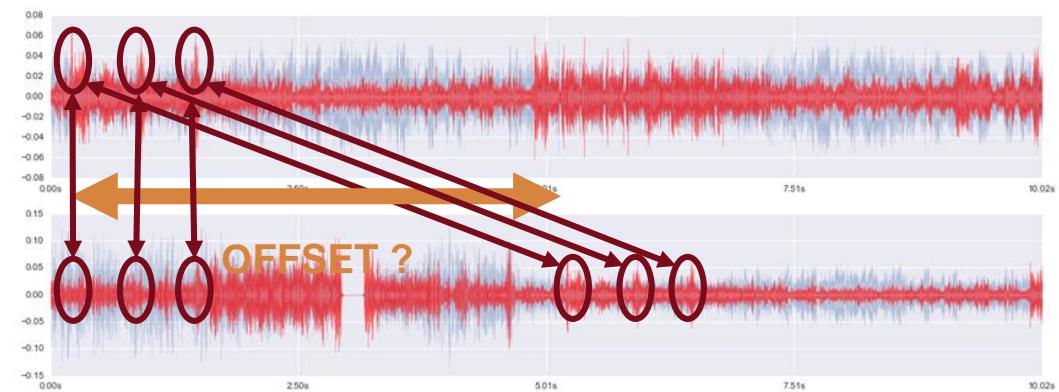
## Task 4 results

- 8 submissions / 7 teams / 23 authors



## Audio-Synchronisation

- Time-Information in video-metadata often unreliable (e.g. personal camera)
- Salient, loud acoustic events create a specific pattern
  - Z.B. honking, screams, gunshots, etc.
  - These are independant from: angle of view, camera position
- Synchronisation by overlapping audio patterns



# Audio and Music related Research at IFS

# Similarity Search



## Search for similar sounding Audio Content in Europeana

- Select a favoured track
- System analyzes audio content
- Provides a list of tracks with calculated similar acoustic properties

The screenshot shows the Europeana search interface with the query "mozart". The results page displays 24 items per page from a total of 14,716 results. Each result card includes a thumbnail, title, and a play button. Below the search bar, there are filters for media type (IMAGE, SOUND, TEXT, VIDEO), language, year, provider, and data source.

The screenshot shows a SoundCloud track page for "Beautiful Moons Ago" by Rachel J. Rodgers. It features a play button, the track title, and a waveform visualization. To the right, there is a video thumbnail showing two women performing. Below the video, there is a comment section and a sidebar for related tracks.

## Search for similar sounding Soundcloud tracks in Europeana

- Supply a Soundcloud URL
- System downloads and analyzes track
- Provides a list of similar sounding Europeana tracks

# Audio-to-Score Alignment

## 1. Extract Music Symbols from Image



Applying Optical Music Recognition

- Coordinates of staves, measures, etc
- Extract Note values

## 2. Feature Extraction

- Make both sides comparable
- Map music energy to the 12 semitones of an Octave
- Can be calculated from symbolic and recorded music

## 2.1 Symbolic Music Feature Extraction



## Synchronization

- Compare symbolic with audio features
- Find matching vectors

## Digitized Score Image

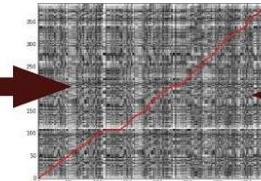


## 4. Map Coordinates from Image to Time-Events of the Audio

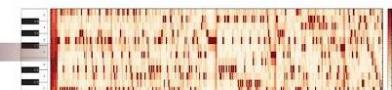
Time	X	Y	Width	Height	Page
0:00	20	120	90	21	1
0:05	110	122	92	20	1
0:09	202	121	91	21	1
0:13	21	162	120	22	1
0:16	141	163	122	20	1

- Create a mapping of time-codes to page coordinates
- Describes which region of the image has to be highlighted at which time

## 3. Align Score with Audio Information



## 2.2 Audio Feature Extraction



## Audio Recording



# Score Following



**Partita BWV 1013**  
**flute solo**

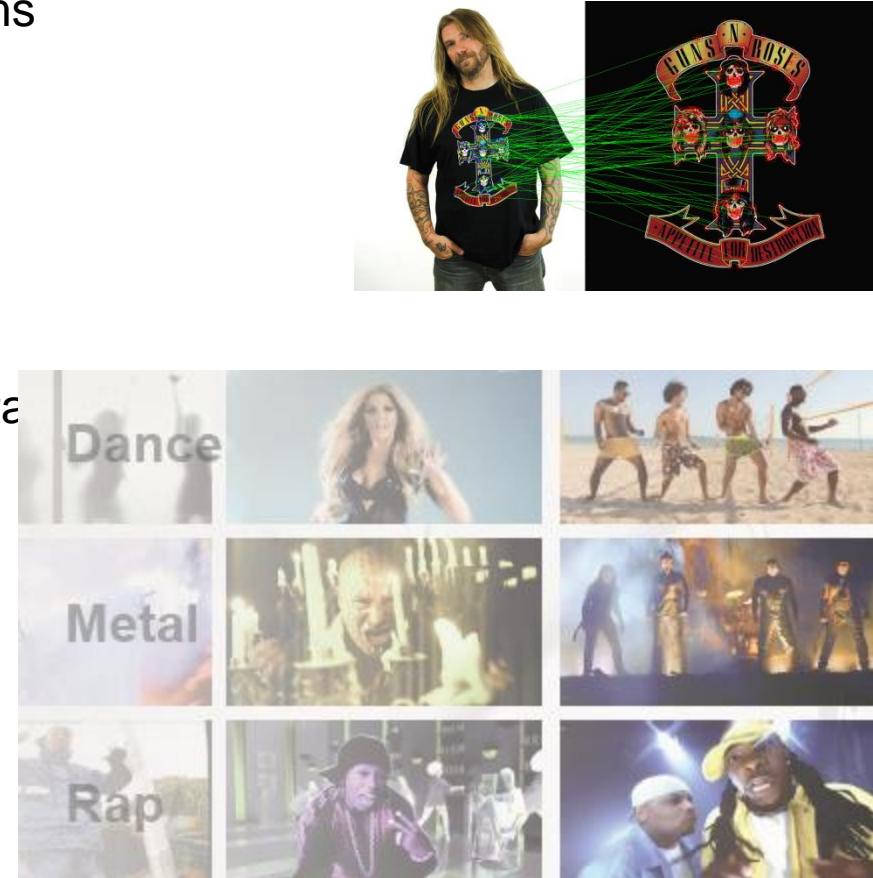
Johann Sebastian Bach  
 typeset by Michele Giulianini

Allemande



[http://www.ifs.tuwien.ac.at/~schindler/files/eusounds/scorefollowing/SFP\\_Bach\\_BWV\\_1013.html](http://www.ifs.tuwien.ac.at/~schindler/files/eusounds/scorefollowing/SFP_Bach_BWV_1013.html)

- Multimodal Approach to MIR Problems
  - Classification / Tagging
  - Mood estimation
  - Music Similarity Retrieval
- Hypothesis:
  - visual layer of music videos contains related information
- Research Aims
  - Can this information be used?
    - Improve MIR solutions
    - Use images as queries



## Top concepts of music video frames examples



stage	0.3162
electric guitar	0.1169
bassoon	0.0649
accordion	0.0611
drumstick	0.0386
microphone	0.0313
marimba	0.0276

mosquito net	0.0932
wardrobe	0.0857
brassiere	0.0815
shower curtain	0.0471
candle	0.0400
plastic bag	0.0204
hoop skirt	0.0187

maillot	0.2745
bolo tie	0.0732
Windsor tie	0.0550
letter opener	0.0486
brassiere	0.0390
bikini	0.0384
bassoon	0.0364

lumbermill	0.1925
tow truck	0.1215
harvester	0.1152
thresher	0.0513
jeep	0.0484
half track	0.0473
pickup truck	0.0460

## Classification results (visual concepts only)

(c) High-level Visual Concepts												
$v_{in1}$	MEAN	1000	66.86	42.09	53.69	51.26	31.23	37.05	46.87	23.90	33.07	
$v_{in2}$	STD	1000	69.78	46.76	50.08	51.95	29.99	32.88	48.29	26.83	29.63	
$v_{in3}$	MAX	1000	73.15	44.26	46.41	54.60	33.05	31.94	50.07	26.93	27.49	
$v_{in4}$	$v_{in3}+v_{in2}$	2000	73.61	46.53	51.21	55.04	31.48	34.00	51.30	27.03	31.04	
$v_{in5}$	$v_{in3}+v_{in1}$	2000	74.36	47.70	53.65	55.99	33.70	37.83	51.58	28.88	33.83	

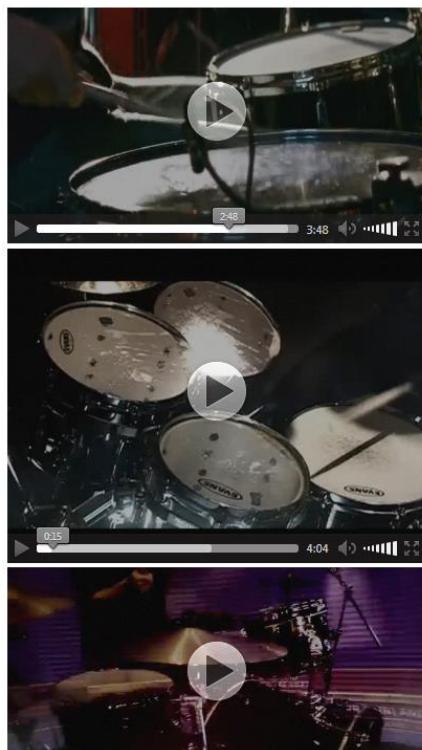
## Semantic Video Search

Search Videos by Objects

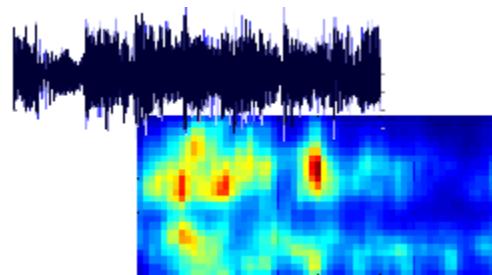
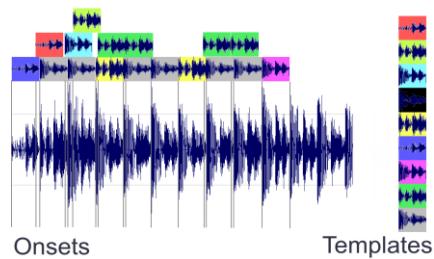
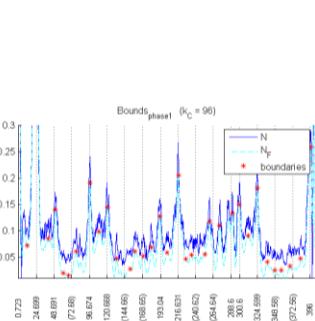
Search for Objects and use Index in next Input-Textfield

Search Term: drum, membranophone, tympan

541 drum, membranophone, tympan



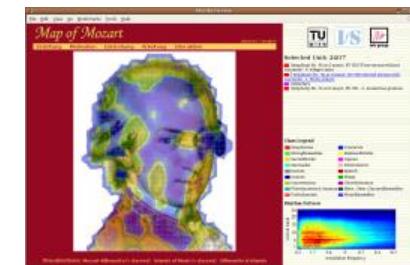
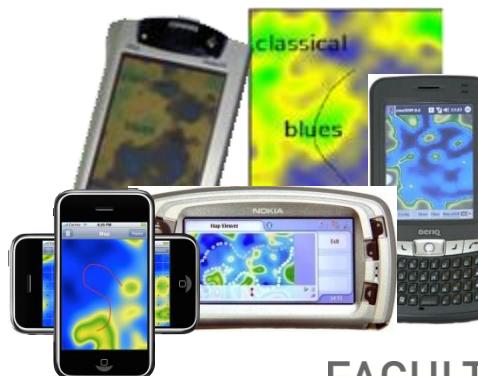
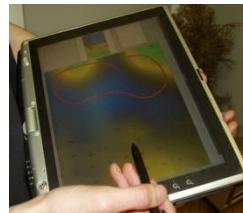
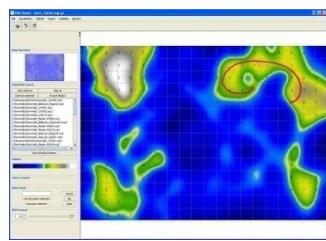
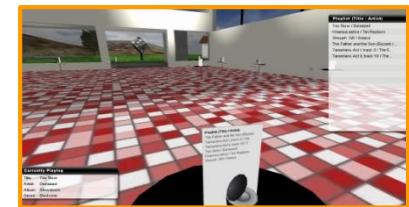
<http://172.20.36.10:5000/>



# Thank You !

Alexander Schindler - schindler@ifs.tuwien.ac.at

<http://www.ifs.tuwien.ac.at/mir>



FACULTY OF **INFORMATICS**