

## CCRMA MIR Workshop 2011 Day 3

Stephen Travis Pope and Steve Tjoa  
Imagine Research  
{stephen,steve}@imagine-research.com

1

## Day 3 Overview

- Overview
- 2nd-Stage Processing
- Segmentation of music and non-musical audio
- Post-processing: What are we doing?
- Classification: KNN vs SVM training and testing
- Clustering vs Classification: Tree-based systems
- Audio Transcription: Onsets and per-onset features
- Other applications Classification/estimation in the presence of polyphony
- Applications



2

## 2nd-Stage Processing

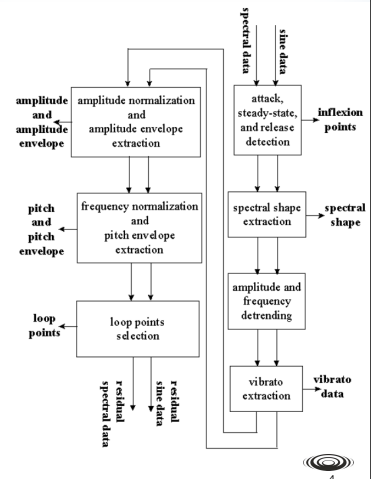
- Thresholds and Data Pruning
- Perceptual Mapping
- Data Reduction
  - Averaging, GMMs, Running Averages
- Feature-data-smoothing
  - De-spiking, sticky values, filter, etc.



3

## Multi-stage Processing

- Feature extraction
- Perceptual mapping
- Signal statistics
- Calculus
- Smoothing
- Segmentation
- Classification



4

## Multi-pass Cross-domain

- Initial segmentation, ID of significant notes or sections based on time-domain or spectral track statistics
- Detailed analysis of short segments (see Fig 1 above to find first note of vocal)
- Suggests heuristic approach, agents, or blackboard architecture



5

## Derived Features

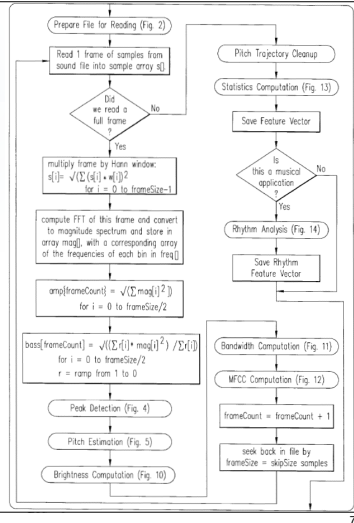
- Adding semantics to low-level features
- Perceptual mapping
- Tracking tempo, meter, harmonic center
- Segmentation: lengths, regularity, intro, coda, fade, per-segment averages
- Instrument identification (signatures)
- Location of important sections/notes



6

## Example

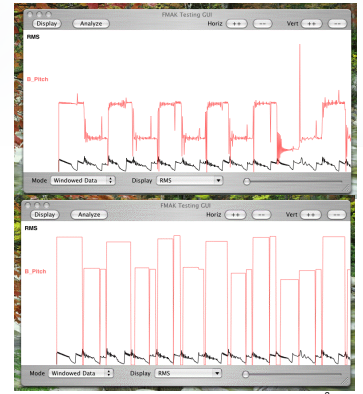
- Complex feature vector processing (from Blum et al. patent #5,918,223)



7

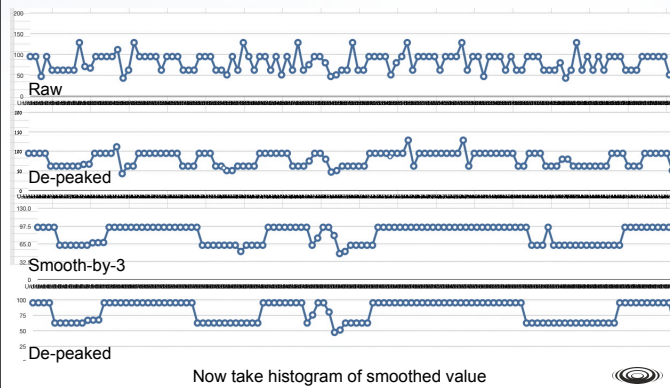
## Data Smoothing and Reduction

- Windowing and processing artifacts
- Use statistical processing and perceptual limits
- Example: bass line for 12-bar blues, before and after smoothing
- Techniques
  - Leaky integrators
  - Sticky values and islands



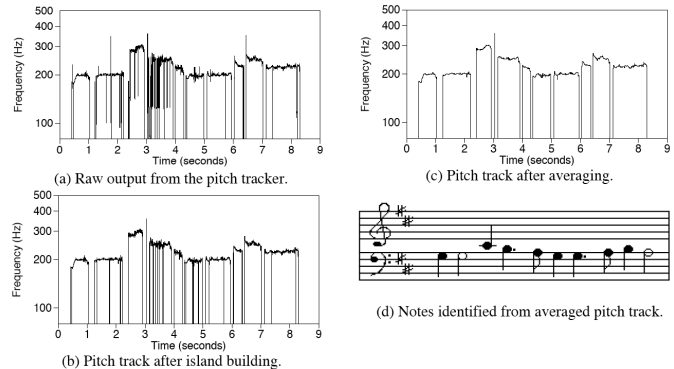
8

## Smoothing (of tempo)



9

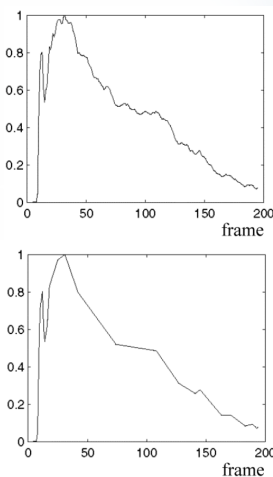
## Typical Cleaning Products



10

## Envelope Data Reduction

- Curve-fitting algorithms
- Least-squared error calculation
- Hierarchical techniques
- Geometrical techniques
- Catalog-based systems



11

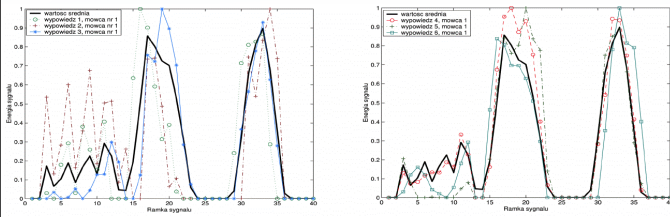
## Perceptual Mapping

- Amplitude
  - dB scale, mapped by equal-loudness curves, grouped into freq. regions
- Frequency
  - Pitch-mapped, scale warped, spectral bands
- Envelopes
- Tempo
- See more examples

12

## Noise and Variability

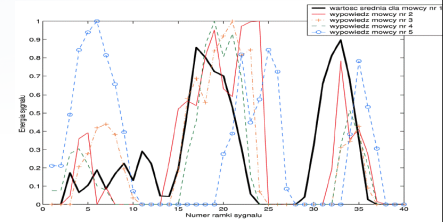
- Energy/time for word “pusc”
- Same speaker, multiple recordings



13

## Variability between Speakers

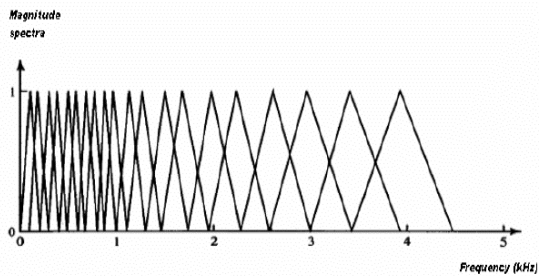
- “Pusc”
- 5 speakers
- Avg. shown in black
- Common features?



14

## Frequency Regions and Scaling

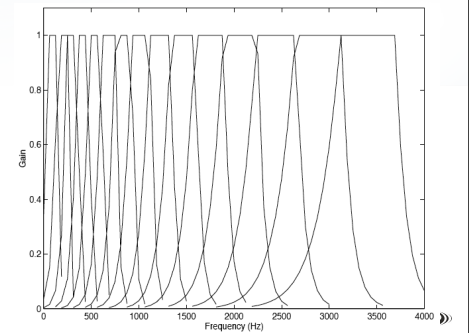
- Mel-warped frequency bands



15

## Bark-scale Trapezoidal Filters

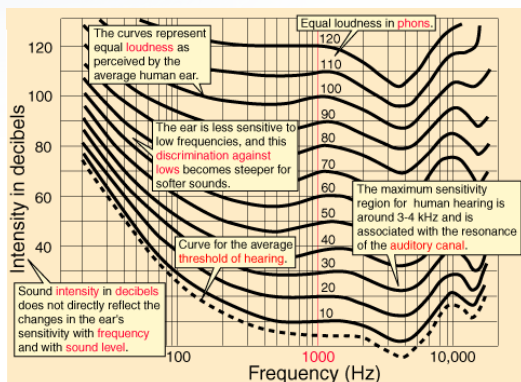
- Similar spacing to Mel filters



16

## Equal-loudness Curves

- Fletcher–Munson vs. Robinson–Dadson



17

## Segmentation of Music and Non-musical Audio

- Segmentation based on islands of similar features
- Segmentation based on regular difference peaks
- Segmentation based on labeling



18

## Audio Segmentation

- Typical audio segmentation requirements
  - Speech vs. music vs. other sound
  - Speaker change detection
  - Discover song verses/refrains
- Segmentation of Music
  - Assume somewhat regular segment length (verse)
  - Relationship to musical form guidelines
  - Assume good autocorrelation of segments
  - What feature weighting to use?



19

## Basic Segmentation Techniques

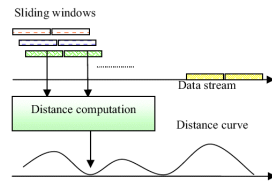
- Inter-window distance metrics
  - Choose a weighting of the feature vector that gives a good dynamic range (of distance)
  - Look for regular distance peaks (autocorrelation of distance metric)
  - Iterate as necessary
- Heuristic techniques
  - Neural nets, HMM
- Statistical techniques
  - Bayesian, Gaussian



20

## Segmentation with a Distance Metric

- Use overlapping windows
- Choose distance metric (feature weighting) carefully or try several of them
- Locate distance peaks, possibly assume they lie at regular intervals
- Examples: speech, single instrument, mixed songs



21

## Distance Metric Weightings

```
# Spectral-/pitch-centric configuration
SegmenterConfiguration {
    kHPRMS 0.5
    kDynamicRange 0.5
    kZeroCrossings 0.5
    kBassPitch 0.5
    kSpectralSlope 1
    kSpectralCentroid 0.5
    kSpectralVariety 1
    kSpectralBandMax 0.5
}

# RMS-centric configuration
SegmenterConfiguration {
    kRMS 1
    kLPRMS 1
    kHPRMS 1
    kSpectralVariety 0.5
    kSpectralBandMax 0.5
    kZeroCrossings 0.5
    kBassPitch 0.5
    kDynamicRange 1
    kTempo 0.5
}

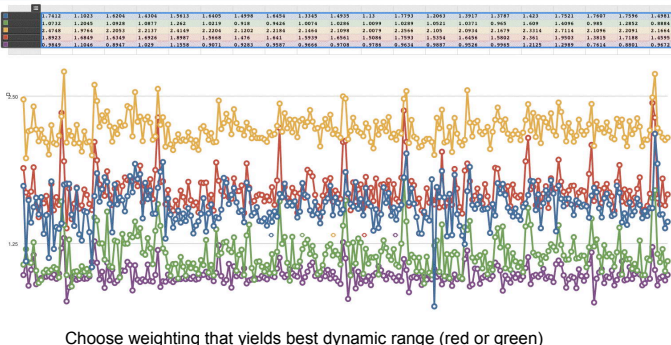
# MFCC- & tracking-centric configuration
SegmenterConfiguration {
    kHPRMS 0.2
    kSpectralVariety 1
    kBassDynamicity 1
    kZeroCrossings 0.2
    kBassPitch 0.5
    kSTrackBirths 0.5
    kSTrackDeaths 0.5
    kMFCCMax 1
    kMFCCFirst 1
    kMFCCAvg 0
}

# Tracking-centric configuration
SegmenterConfiguration {
    kRMS 0.5
    kHPRMS 0.2
    kSpectralSlope 0.2
    kSpectralCentroid 0.2
    kSpectralVariety 0.5
    kSpectralBandMax 0.5
    kBassPitch 1
    kSTrackBirths 1
    kSTrackDeaths 1
    kMFCCFirst 1
    kTempo 0.5
}
```



22

## Distance Metrics with 5 Weightings



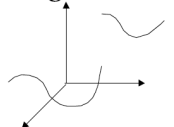
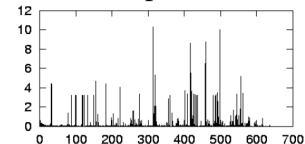
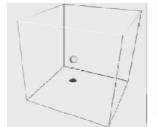
Choose weighting that yields best dynamic range (red or green)



23

## Time-based signal segmentation (CT)

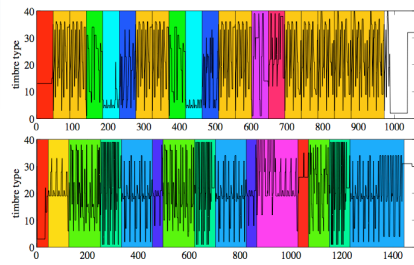
- > Time series of feature vectors  $V(t)$
- >  $f(t) = d(V(t), V(t-1))$ 
  - $d(x,y) = (x-y)C^{-1}(x-y)^t$  (Mahalanobis)
- >  $df/dt$  peaks correspond to texture changes



24

## Segmentation Examples

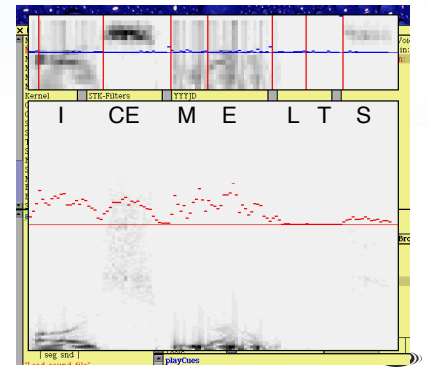
- SoundBite finds salient segments
- 1/8 octave Constant Q transform, PCA, HMMs



25

## Segmentation Examples

- 8S (Siren/Smalltalk) phoneme segmenter
- 1/2-octave filter banks, RMS, LPC coeff



26

## Song Segmentation Data

```
select artist, title, SegmentWeight, NumSegments, VerseLength,
       TypicalStart, SoloStart, SoloCentroid, SoloVariety, SoloTempo,
       SoloDynRange from FSongs where title = 'I Believe In Love';
```

| artist     | title             | segmentweight | numsegments |
|------------|-------------------|---------------|-------------|
| Paula Cole | I Believe In Love | 0.923772      | 0.24 (7)    |

| verselength | typicalstart | solostart |
|-------------|--------------|-----------|
| 0.631119    | 0.280232     | 0.590672  |

| s_centroid | s_variety | s_tempo | s_dynrange |
|------------|-----------|---------|------------|
| 0.4991     | 0.001422  | 0.3360  | 0.654455   |



27

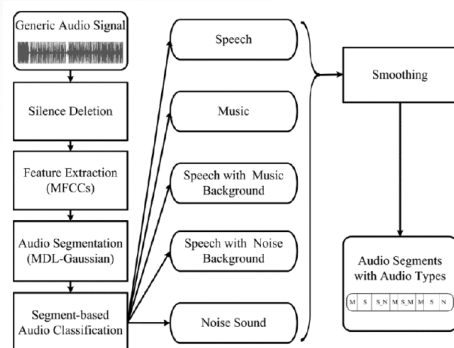
## Rhythmic Segmentation

- Windowed amplitude envelope, extract note onsets if possible
- Look for changes in total energy (e.g., silence)
- Look for regularly-space events at several levels of hierarchy (assume ratios of 2 or 3 as basis)
- Works well for many genres



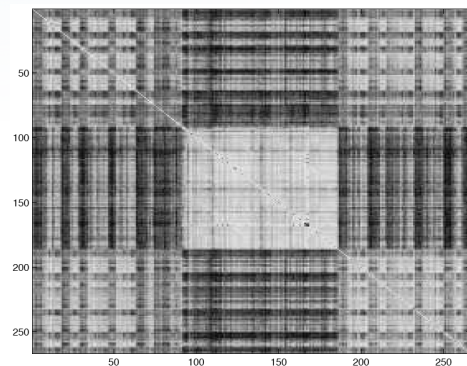
28

## Segmentation/Classification Framework



29

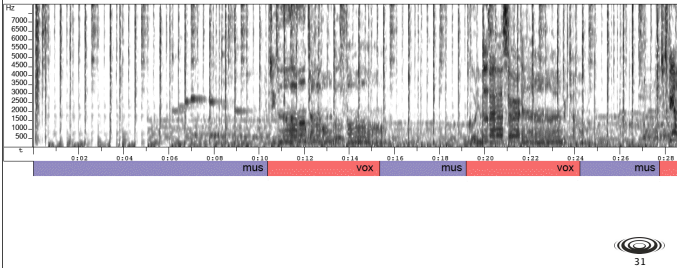
## Self-Similarity and Recurrence Maps



30

## Segmentation for Indexing

- Berenzweig's singing detector



31

## Segmented Feature Vectors

- Multi-stage feature extraction
  - 1: Data analysis and segmentation (using a complex set of distance metrics)
  - 2: Prune the data to maintain a small set of feature vectors (1-per-segment & song peak/average data).
- Add in segment-length statistics, confidence
- Delivers a complex feature matrix with segment-length data, segmentation confidence, and a collection of individual feature vectors

32

## Using Segmentation

- Seg for summarization
- Seg for clustering
- Seg for finger-printing
- Seg statistics alone
  - # segs, confidence, fade I/O
- Song structure
  - typical/solo verses
- Solo instr, singer ID

33

## Post-processing

- What are we doing?
- Storing Feature Data
  - SQL, JSON, XML, etc.
- Classification/Clustering
- Transcription
- Labeling
- Mixed-mode apps

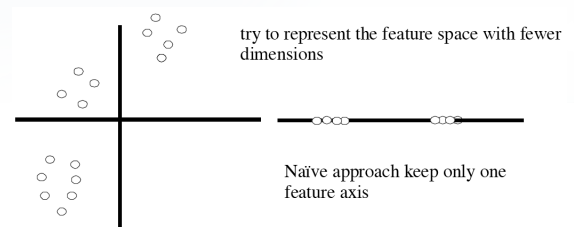
34

## Processing Large Data Sets

- Given a rich feature vector for a large data set
  - Dimensionality reduction – find a small set of discriminating variables or feature weightings
  - Clustering – find groups in a higher-dimensional space
- (Related?) Classify a sound according to a set of criteria and/or a catalog

35

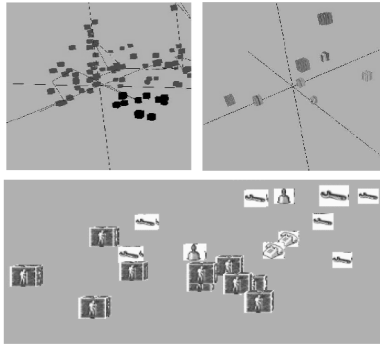
## Dimensionality Reduction



36

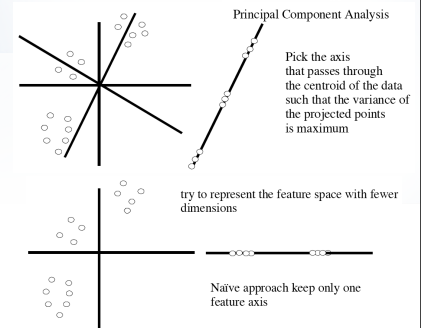
## Dimensionality Reduction

- Grouping of large data sets
- PCA & ICA
- KNN and clustering



## PCA

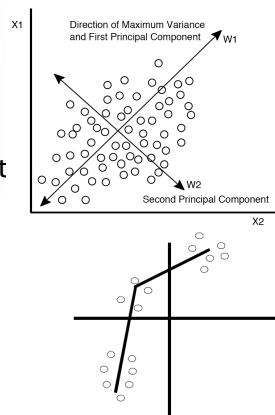
- Find the principal components or axes of discrimination of a set of  $n$ -dimensional data points



AKA Karhunen-Loève transform (or KLT) or data whitening

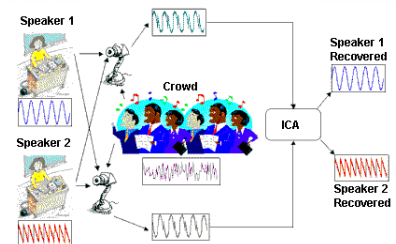
## PCA

- Easy with clear clusters or equal density
- Often necessary to repeat with different weighting metrics to find good clustering criteria
- Statistics based on covariance matrices



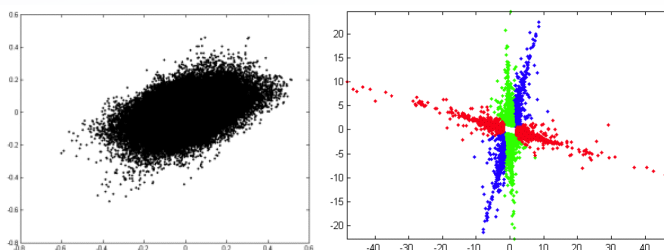
## Independent Component Analysis (ICA)

- Relative of non-linear PCA
- Discovers independent basis functions
- Effective for blind source separation



## PCA Examples

- Same feature vectors, different weightings
- 90% of FV is "noise"



## Problems Working in High-dimensional Spaces

- „Curse of dimensionality“
  - High overlap of MBRs in almost all directory nodes
  - Hence, almost all (data) pages have to be scanned
  - Query time approaches  $O(n)$ : sequential scan becomes competitive
- High-D and nearest neighbors
  - Small statistical variance for distances to other points in data base
  - Is „nearest neighbor“ meaningful in high-D spaces? [BGRS 99]
  - Reasonability of NN depends on data distribution [Keim et al.]
  - Real data are typically not distributed evenly but form clusters
- Techniques to cope with problems of high-D
  - Reduction of dimensionality
  - Accelerating the sequential scan (VA-File, IQ-tree, ...)

## Dimensionality Reduction

### Linear Reduction Techniques

#### First phase: Rotation of data space

- Principal Components Analysis (PCA, KLT):  $O(d^3) + O(d^2)$
- Discrete Fourier/Cosine Transform:  $O(d \log d)$
- Wavelet Transform:  $O(d)$

#### Second phase: Projection to lower dimensions ( $d \rightarrow r$ )

- After rotation, just discard the last  $(d - r)$  coordinates
- $L_p$  distance values for sure get smaller: 
$$\left( \sum_{i=1}^r |p_i - q_i|^p \right)^{1/p} \leq \left( \sum_{i=1}^d |p_i - q_i|^p \right)^{1/p}$$

### Non-linear techniques

- Multi-Dimensional Scaling (MDS)
- FastMap (extension of MDS) [Fal+]



43

## Example: Feature Rank (InfoGain)

|           |                       |           |                    |
|-----------|-----------------------|-----------|--------------------|
| - 0.22089 | 8 SpectralVariety     | - 0.06263 | 44 fp_gravity      |
| - 0.20219 | 55 SpectralVarietyVar | - 0.05001 | 12 SpectralBandMax |
| - 0.19994 | 46 fp_focus           | - 0.03816 | 38 HighPeakAmp     |
| - 0.17689 | 20 MFCCCoeff4         | - 0.03233 | 41 BHSUM1          |
| - 0.16469 | 18 MFCCCoeff2         | - 0.02625 | 37 LowPeakBPM      |
| - 0.16457 | 21 MFCCCoeff5         | - 0.02625 | 36 LowPeakAmp      |
| - 0.16404 | 19 MFCCCoeff3         | - 0.02625 | 40 HighLowRatio    |
| - 0.12209 | 22 MFCCCoeff6         | - 0.02625 | 39 HighPeakBPM     |
| - 0.12151 | 17 MFCCCoeff1         | - 0.02455 | 35 SoloRMS         |
| - 0.11668 | 45 fp_bass            | - 0.02414 | 34 SoloTempo       |
| - 0.08726 | 43 BHSUM3             | - 0.02105 | 42 BHSUM2          |
| - 0.08701 | 24 STrackDeaths       | - 0.02001 | 33 SoloDynRange    |
| - 0.08701 | 23 STrackBirths       | - 0.01918 | 9 SpectralFlux     |
| - 0.08701 | 16 SpectralBand4      | - 0.01775 | 15 SpectralBand3   |
| - 0.08475 | 7 SpectralSlope       | - 0.01772 | 14 SpectralBand2   |
| - 0.08387 | 10 SpectralRollOff    | - 0.01715 | 13 SpectralBand1   |
| - 0.08115 | 51 ZeroCrossingsVar   | - 0.01026 | 25 TempoAvg        |
| - 0.07597 | 4 ZeroCrossings       | - 0.01001 | 32 SoloVariety     |
| - 0.07    | 2 LPRMS               | - 0.00991 | 31 SoloCentroid    |
| - 0.06621 | 3 HPRMS               |           |                    |
| - 0.06447 | 47 RMSVar             |           |                    |



44

## Example: 1st 2 PCA Dimensions

- 0.18 MFCCCoeff6 + 0.18 MFCCCoeff5 + 0.18 MFCCCoeff4 + 0.18 MFCCCoeff3 + 0.18 MFCCCoeff2 + 0.18 SpectralFlux + 0.18 SpectralRollOffVar + 0.18 SpectralSlope + 0.18 MFCCCoeff6Var + 0.18 MFCCCoeff5Var + 0.18 MFCCCoeff4Var + 0.18 MFCCCoeff3Var + 0.18 MFCCCoeff2Var + 0.179 SpectralBand2Var + 0.179 SpectralBand1Var + 0.179 STrackBirthsVar + 0.179 STrackDeathsVar + 0.179 SpectralBand4Var + 0.179 SpectralBand3Var + 0.179 SpectralBand3 + 0.179 SpectralBand2
- 0.378 BHSUM3 - 0.345 LowPeakAmp - 0.323 BHSUM1 - 0.309 BHSUM2 - 0.298 HighPeakAmp - 0.294 fp\_bass - 0.269 ZeroCrossingsVar - 0.267 ZeroCrossings - 0.237 HPRMS + 0.161 TempoWeight + 0.161 TempoAvg - 0.15 fp\_gravity - 0.12 LPRMSVar + 0.114 HighPeakBPM + 0.114fp\_focus + 0.093 LowPeakBPM + 0.086 HPRMSVar - 0.058 SpectralVarietyVar + 0.054 QuietSections + 0.053 LPRMS - 0.049 SpectralBandMaxVar - 0.049 RMSVar - 0.045 PeakVar - 0.041 SoloTempo



45

## Example: \*ART-Tree-learning

```

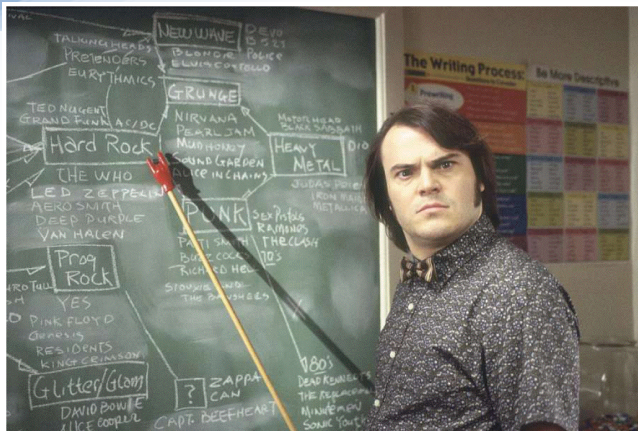
SpectralVarietyVar <= 0.021884
| fp_focus <= 0.415299
| | MFCCCoeff1Var <= 0.127492
| | | fp_bass <= 0.412635
| | | | BassDynamicity <= 0.698656
| | | | | DynamicRangeVar <= 0.370633
| | | | | | SpectralCentroidVar <= 0
| | | | | | | HPRMS <= 0.867819: Rock-Alternative (17.0/3.0)
| | | | | | | | HPRMS > 0.867819
| | | | | | | | | SoloStart <= 0.015131: Rock (3.0)
| | | | | | | | | | SoloStart > 0.015131: Rock-Alternative (3.0/1.0)
| | | | | | | | | | | SpectralCentroidVar > 0: Rock (4.0/2.0)
| | | | | | | | | | | | DynamicRangeVar > 0.370633: Rock (5.0/1.0)
| | | | | | | | | | | | | BassDynamicity > 0.698656
| | | | | | | | | | | | | | LPRMS <= 0.852023: unknown (2.0/1.0)
| | | | | | | | | | | | | | | LPRMS > 0.852023: Pop-BritPop (2.0/1.0)
| | | | | | | | | | | | | | | | fp_bass > 0.412635
| | | | | | | | | | | | | | | | | LPRMS <= 0.687344: Comedy (2.0/1.0)
| | | | | | | | | | | | | | | | | | LPRMS > 0.687344: unknown (2.0/1.0)
| | | | | | | | | | | | | | | | | | MFCCCoeff1Var > 0.127492
| | | | | | | | | | | | | | | | | | | FadeOut <= 0.1
| | | | | | | | | | | | | | | | | | | | NumSegments <= 0: Jazz-Big Band_Swing (3.0/1.0)
| | | | | | | | | | | | | | | | | | | | NumSegments > 0
| | | | | | | | | | | | | | | | | | | | | SoloVariety <= 0.001143: Blues (3.0/1.0)
| | | | | | | | | | | | | | | | | | | | | SoloVariety > 0.001143: Rock (3.0/1.0)

```



46

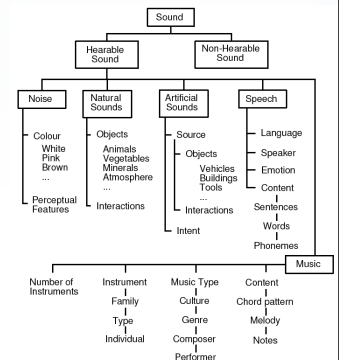
## Music Classification



47

## Classification

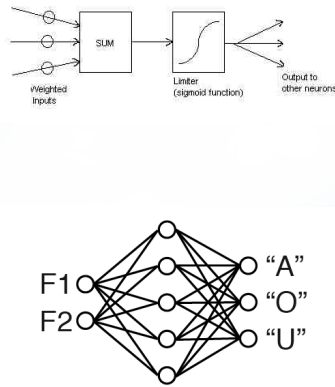
- Data clustering in high-dimensional non-linear spaces is complex
- Mapping discovered clusters to genres isn't obvious
- Rule-based techniques
- Easy for a small number of genres ( $< 10$ )
- Quite a challenge for many genres ( $> 50$ )
- Requires a large segmented feature vector



48

# Neural Networks

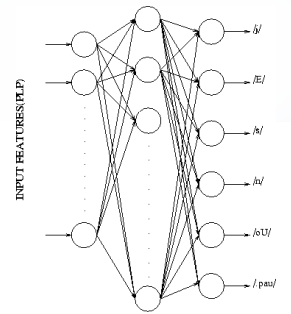
- Traditional 3-layer weighted NN model
  - Input
  - Hidden
  - Output
- Training by back-propagation of error
- Used for classification



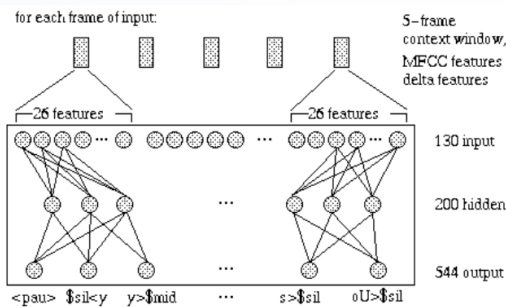
49

## NN Training

- Given inputs and desired outputs,
- Iterate tuning neuron weights to distribute errors
- Relation to simulated annealing, weighted statistical networks, etc.



## NN/NLP Example



- ♦ trained on: OGI Numbers, Yes/No, Apple, and Stories corpora, and NYNEX PhoneBook corpus.



51

## Classification Confusion Matrices

- Off-axis values = misclassification

|           | Classical | Country | Disco | Hiphop | Jazz | Rock | Blues | Reggae | Pop | Metal |
|-----------|-----------|---------|-------|--------|------|------|-------|--------|-----|-------|
| Classical | 73        | 0       | 0     | 0      | 6    | 2    | 0     | 0      | 0   | 0     |
| Country   | 0         | 43      | 1     | 0      | 1    | 6    | 2     | 4      | 3   | 1     |
| Disco     | 0         | 6       | 43    | 10     | 0    | 4    | 9     | 3      | 3   | 2     |
| Hiphop    | 0         | 4       | 9     | 42     | 0    | 3    | 2     | 17     | 10  | 2     |
| Jazz      | 21        | 5       | 1     | 0      | 21   | 6    | 5     | 0      | 2   | 3     |
| Rock      | 4         | 16      | 6     | 1      | 8    | 41   | 11    | 5      | 11  | 17    |
| Blues     | 2         | 18      | 2     | 1      | 7    | 7    | 61    | 5      | 1   | 2     |
| Reggae    | 0         | 2       | 12    | 30     | 2    | 13   | 6     | 63     | 6   | 0     |
| Pop       | 0         | 3       | 25    | 8      | 3    | 7    | 0     | 3      | 64  | 2     |
| Metal     | 0         | 3       | 1     | 1      | 1    | 11   | 4     | 0      | 0   | 71    |



52

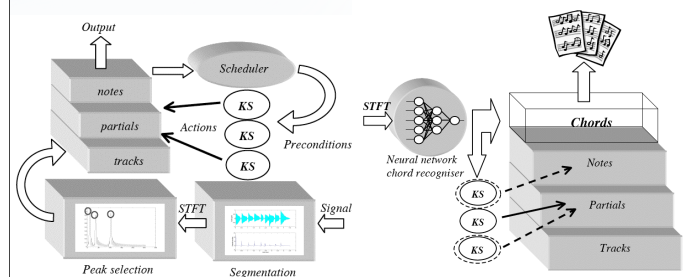
# Markov Tables and Models

- **Background**
  - Transition tables
  - FSA-like “chains”
- These can be learned
  - E.g., pitch set TTs can be used as composer discriminators for classification
- There can be several TTs in a model (HMMs)

|    | A   | AB  | B    | C   | D   | E   | F   | G   | GH   |
|----|-----|-----|------|-----|-----|-----|-----|-----|------|
| A  | 419 |     | 319  |     | 219 | 119 |     | 619 | 319  |
| AB | 1   |     |      |     |     |     |     |     |      |
| B  | 215 |     | 1115 | 415 |     | 315 |     |     |      |
| C  |     |     | 615  | 315 | 615 |     |     |     |      |
| D  |     |     |      | 311 | 311 | 511 |     |     |      |
| E  | 419 | 119 |      | 319 |     | 519 | 419 | 119 | 1119 |
| F  |     |     | 15   |     |     | 15  |     | 35  |      |
| G  | 15  |     | 15   | 25  |     |     |     | 15  |      |
| GH |     |     | 34   |     |     | 14  |     |     |      |



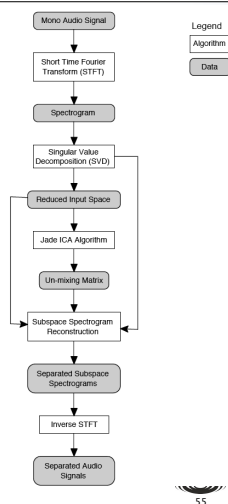
## Bello's Examples



54

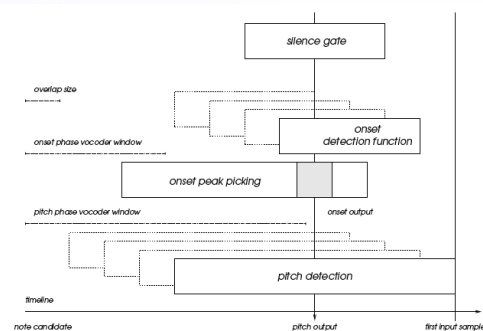
## Multi-stage Systems (again)

- Incorporate statistical and heuristic methods (possibly using large data stores) after cross-domain multi-stage feature extraction



55

## Windowed Multi-stage



56

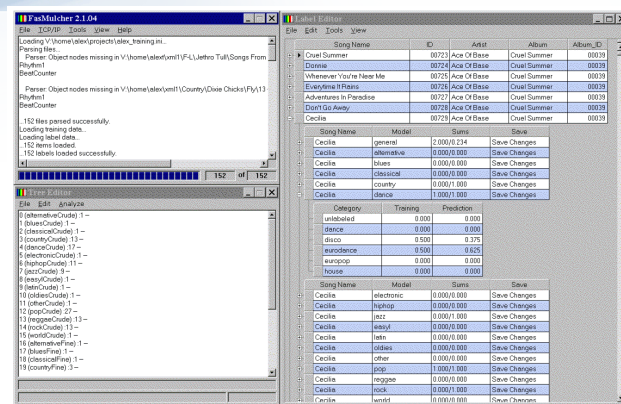
## Clustering vs Classification

- Tree-based systems
  - CART
- Partitioning System
  - CURE



57

## Regression Trees & CART



58

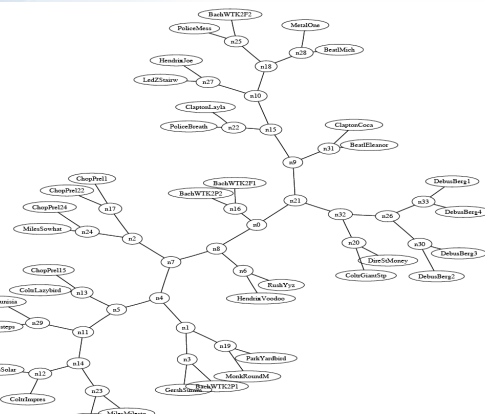
## Rule-learning

- $fp\_focus > 0.519652$
- $AND \quad fp\_bass \leq 0.273051$
- $AND \quad LPRMS \leq 0.858613$
- $AND \quad ZeroCrossings \leq 0.197117$
- $AND \quad FadeOut > 0.1 \quad AND \quad FadeOut \leq 0.7$
- $AND \quad SpectralVarietyVar \leq 0.021011$
- $AND \quad MFCCCoeff1Var \leq 0.393386$
- $AND \quad SpectralBandMaxVar \leq 0.003615$
- $--> \text{Classical } (78.0/1.0)$



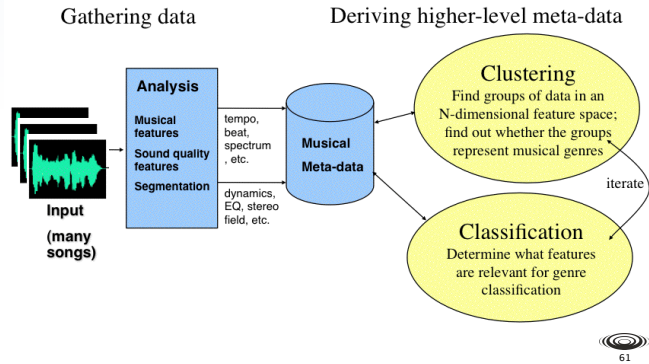
59

## Algorithmic Clustering



60

## Statistical Methods for Clustering and Classification



61

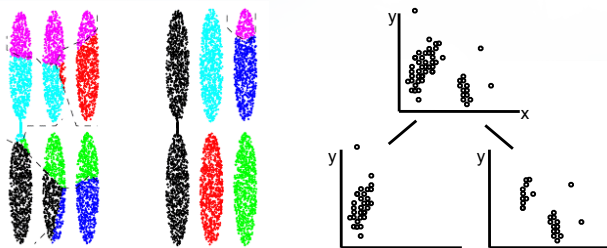
## Clustering and Classification

- Clustering: Given a feature vector derived for a data set, find the feature weighting for which the data groups are best delineated
- Classification: relate clusters to perceptual groups such as speech/music/snd-effect or musical genres



62

## Clustering Errors



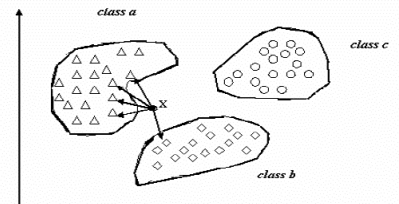
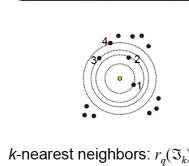
63

## K Nearest Neighbor Clustering

- For each item, locate its k nearest neighbors and collect the distances to them
- Apply a threshold to identify data clusters
- Change distance metric as necessary

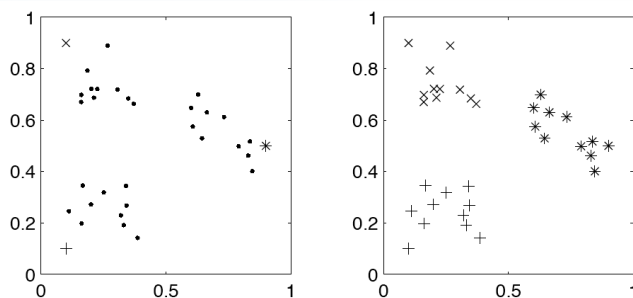
$$d_q\text{-Ranking } r_q: \mathcal{I}_{DB} \rightarrow DB$$

$$i_1 \leq i_2 \Rightarrow d(r_q(i_1), q) \leq d(r_q(i_2), q)$$



64

## 2-Stage KNN Clustering



65

## Advanced KNN/Region Systems

- Hierarchical indexes (trees) for high dimensions
  - Adopt the size of directory nodes in order to locally simulating sequential scans of the data
  - Examples: X-Tree [BKK 96], IQ-Tree [BBJK+ 00]
- Metric Indexes for spaces w/o dimensions
  - For arbitrary distance functions (edit dist, morphological dist)
  - Applicable even to high-dimensional spaces
  - Examples: M-Tree [CPZ 97], Slim tree [FSTT 00]
- Anchoring Methods (vantage points, etc.)
  - Select  $r$  landmarks among the data objects
  - Use distance values to  $r$  landmarks as  $r$  dimensions
  - Examples: vp-tree [FCCM 00], mvp-tree [TÖ 97], [VV 02]

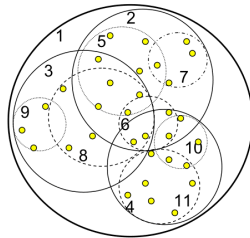


66

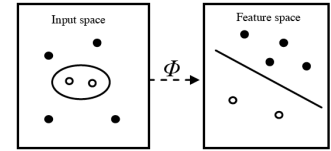
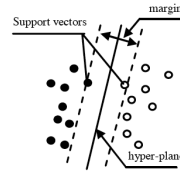
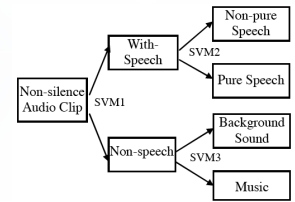
## Region and Sphere Trees

- Many variants of N-D KNN algorithms

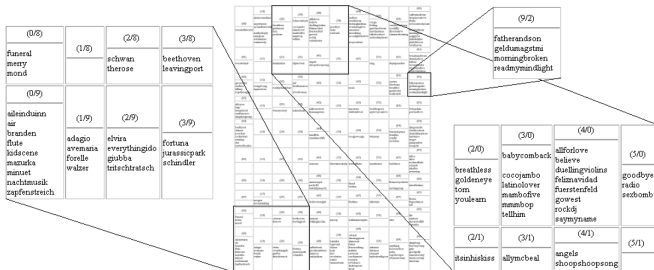
- Sphere-organized Tree
  - R-tree like balanced directory with overlapping regions
  - Use minimum bounding spheres (MBS) instead of MBRs



## Support-Vector Machines



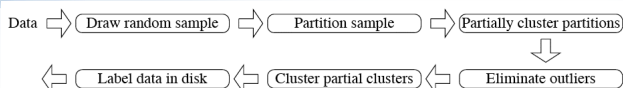
## Self-Organizing Maps



## Clustering Using Representatives (CURE)

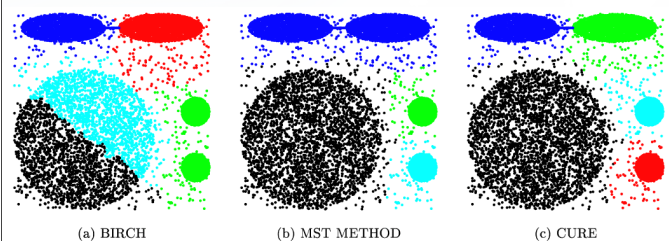
- Middle path between single representative (centroid based) or full representative (complete linkage)
- Uses multiple representative points to evaluate the distance between clusters, adjusts well to arbitrary shaped clusters and avoids single-link effect
- Pick/label representatives carefully

## The CURE Method

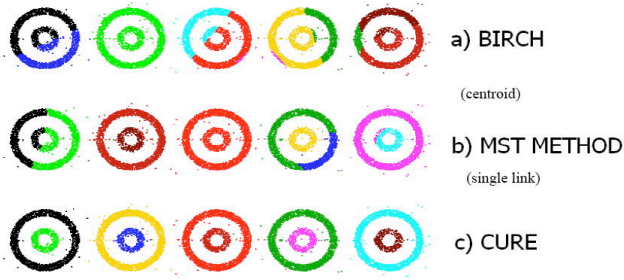


- Draw random sample  $s$ .
- Partition sample to  $p$  partitions with size  $s/p$
- Partially cluster partitions into  $s/pq$  clusters
- Eliminate outliers
  - By random sampling
  - If a cluster grows too slow, eliminate it.
- Cluster partial clusters.
- Label data in disk

## Clustering Results for Various Methods



## Experimental Results: CURE



Picture from CURE, Guha, Rastogi, Shim.



73

## CURE's Alpha Factor

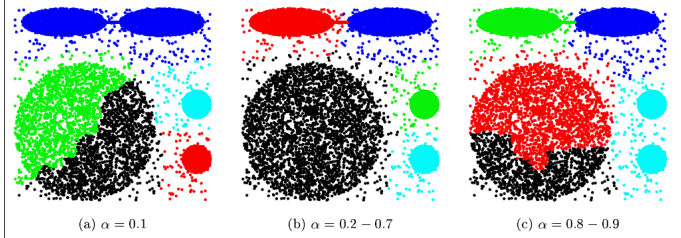


Figure 9: Shrink factor toward centroid



74

## Using CURE

The clusterer command is executed with command-line options to configure the clustering partition pools.

The options are:

- p [NUM] - Number of pre-clustering partitions (2 - 50)
- q [NUM] - Pre-clustering partition factor (3)
- k [NUM] - Desired number of Clusters (4 - 50)
- c [NUM] - Representatives per cluster (2 - 15)
- a [NUM] - Representative scaling factor (0.2 - 0.7)

[OPTIONAL\_FLAGS]:

- x [STR] - configuration file name [Configuration.txt]
- t - Displays the parsed command parameters and exits
- ? - Displays this message

Typical values (for a database of about 50000 entries):

fclusterer -p 20 -q 3 -k 20 -c 8 -a 0.4



75

## CURE Clusterer Execution

```
----- PRE CLUSTERING ROUND -----
Input Space Size : 5000
Partitions : 4
Partition Sizes : 1250
Clusters / Partition : 417

Cluster run #0 clustering 1250 data points...
Cluster run #0 found 541 representatives.
Updating representative data points to new status (541)
Updating non-representative data points status (709)
Cluster run #0 done.

Cluster run #1 clustering 1250 data points...
Cluster run #1 found 486 representatives.
Updating representative data points to new status (486)
Updating non-representative data points status (764)
Cluster run #1 done.

Cluster run #2 clustering 1250 data points...
Cluster run #2 found 517 representatives.
Updating representative data points to new status (517)
Updating non-representative data points status (733)
Cluster run #2 done.

Cluster run #3 clustering 1250 data points...
Cluster run #3 found 517 representatives.
Updating representative data points to new status (517)
Updating non-representative data points status (733)
Cluster run #3 done.

----- CLUSTERING FINAL REPRESENTATIVES -----
Final cluster run clustering 2024 data points...
Final cluster run found 74 representatives.
Number of Clusters constructed : 7024
Number of Clusters destroyed : 7004
Number of DataPoints constructed : 0
Number of DataPoints destroyed : 50927

Updating non-representative data points status(1950)
Marking final cluster 1 with 1 reps
Marking final cluster 2 with 10 reps
Marking final cluster 3 with 10 reps
Marking final cluster 4 with 1 reps
Marking final cluster 5 with 1 reps
Marking final cluster 6 with 3 reps
Marking final cluster 7 with 4 reps
Marking final cluster 8 with 1 reps
Marking final cluster 9 with 4 reps
Marking final cluster 10 with 10 reps
Marking final cluster 11 with 10 reps
Marking final cluster 12 with 1 reps
Marking final cluster 13 with 1 reps
Marking final cluster 14 with 1 reps
Marking final cluster 15 with 1 reps
Marking final cluster 16 with 1 reps
Marking final cluster 17 with 1 reps
Marking final cluster 18 with 1 reps
Marking final cluster 19 with 10 reps
Marking final cluster 20 with 2 reps
end-of final cluster run done.

----- LABELING DB ITEMS -----
Reading cluster member table
Reading cluster representative table
RunLabelingRound on 4926 points with 74 cluster reps

----- POST-PROCESSING DB -----
Run post processing

----- CLUSTERER DONE -----
```



76

## Cross-genre Distances

Mixed - 2 Blondie, 2 Cat Stevens, 2 Bill Cosby  
PCA-semi-weighted EMD

| fdistance | 45851   | 45953   | 36818   | 36680   | 26846   | 26861   |
|-----------|---------|---------|---------|---------|---------|---------|
| 45851     | 0.00000 | 0.18307 | 0.21873 | 0.24543 | 0.24404 | 1.08880 |
| 45953     | 0.18307 | 0.00000 | 0.14053 | 0.20046 | 0.24095 | 1.08670 |
| 36818     | 0.21873 | 0.14053 | 0.00000 | 0.11590 | 0.25070 | 1.09486 |
| 36680     | 0.24543 | 0.20046 | 0.11590 | 0.00000 | 0.27534 | 1.10789 |
| 26846     | 0.24404 | 0.24095 | 0.25070 | 0.27534 | 0.00000 | 1.08433 |
| 26861     | 1.08880 | 1.08670 | 1.09486 | 1.10789 | 1.08433 | 0.00000 |



77

## Example Genre Confusion Matrix

|   | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | aa | ab | ac | ad | ae | af | ag | ah | ai | aj | ak | al | am | an | ao | ap | aq | ar | as | at | au | av | aw | ax | ay | az | ba | bb | bc | bd | be | bf | bg | bh | bi | bj | bk | bl | bm | bn | bo | bp | bq | br | bs | bt | bu | bv | bw | bx | by | bz | ca | cb | cc | cd | ce | cf | cg | ch | ci | cj | ck | cl | cm | cn | co | cp | cq | cr | cs | ct | cu | cv | cw | cx | cy | cz | da | db | dc | dd | de | df | dg | dh | di | dj | dk | dl | dm | dn | do | dp | dq | dr | ds | dt | du | dv | dw | dx | dy | dz | ea | eb | ec | ed | ee | ef | eg | eh | ei | ej | ek | el | em | en | eo | ep | eq | er | es | et | eu | ev | ew | ex | ey | ez | fa | fb | fc | fd | fe | ff | fg | fh | fi | fj | fk | fl | fm | fn | fo | fp | fq | fr | fs | ft | fu | fv | fw | fx | fy | fz | ga | gb | gc | gd | ge | gf | gg | gh | gi | gj | gk | gl | gm | gn | go | gp | gq | gr | gs | gt | gu | gv | gw | gx | gy | gz | ha | hb | hc | hd | he | hf | hg | hh | hi | hj | hk | hl | hm | hn | ho | hp | hq | hr | hs | ht | hu | hv | hw | hx | hy | hz | ia | ib | ic | id | ie | if | ig | ih | ii | ij | ik | il | im | in | io | ip | iq | ir | is | it | iu | iv | iw | ix | iy | iz | ja | jb | jc | jd | je | jf | jj | jh | ji | jk | jl | jm | jn | jo | jp | jq | jr | js | jt | ju | jv | jw | jx | ky | kz | la | lb | lc | ld | le | lf | lg | lh | li | lj | lk | ll | lm | ln | lo | lp | lq | lr | ls | lt | lu | lv | lw | lx | ly | lz | ma | mb | mc | md | me | mf | mg | mh | mi | mj | mk | ml | mm | mn | mo | mp | mq | mr | ms | mt | mu | mv | mw | mx | my | mz | na | nb | nc | nd | ne | nf | ng | nh | ni | nj | nk | nl | nm | nn | no | np | nq | nr | ns | nt | nu | nv | nw | nx | ny | nz | oa | ob | oc | od | oe | of | og | oh | oi | oj | ok | ol | om | on | oo | op | oq | or | os | ot | ou | ov | ow | ox | oy | oz | pa | pb | pc | pd | pe | pf | pg | ph | pi | pj | pk | pl | pm | pn | po | pp | pq | pr | ps | pt | pu | pv | pw | px | py | pz | qa | qb | qc | qd | qe | qf | qg | qh | qi | qj | qk | ql | qm | qn | qo | qp | qq | qr | qs | qt | qu | qv | qw | qx | qy | qz | ra | rb | rc | rd | re | rf | rg | rh | ri | rj | rk | rl | rm | rn | ro | rp | rq | rr | rs | rt | ru | rv | rw | rx | ry | rz | sa | sb | sc | sd | se | sf | sg | sh | si | sj | sk | sl | sm | sn | so | sp | sq | sr | ss | st | su | sv | sw | sx | sy | sz | ta | tb | tc | td | te | tf | tg | th | ti | tj | tk | tl | tm | tn | to | tp | tq | tr | ts | tt | tu | tv | tw | tx | ty | tz | ua | ub | uc | ud | ue | uf | ug | uh | ui | uj | uk | ul | um | un | uo | up | uq | ur | us | ut | uu | uv | uw | ux | uy | uz | va | vb | vc | vd | ve | vf | vg | vh | vi | vj | vk | vl | vm | vn | vo | vp | vq | vr | vs | vt | vu | vv | vw | wx | wy | wz | xa | xb | yc | yd | ze | zf | zg | zh | zi | zj | zk | zl | zm | zn | zo | zp | zq | zr | zs | zt | zu | zv | zw | zx | zy | zz |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |



78

## Audio Transcription

- Onset-detection
  - See above
- Per-onset features
  - See above
- Pruning, selection
  - Rule-based system?



79

## Meter Estimation Systems

Table 1. Characteristics of some meter estimation systems.

| Reference                    | Input       | Output                    | Technique   |
|------------------------------|-------------|---------------------------|---|
| Temperley & Sleator (1999)   | MIDI        | Meter, time quantization  | Rule-based approach; implementation of the preference rules in (Lerdahl et al., 1983) |
| Dixon (2001)                 | MIDI, audio | Tactus                    | First find periods using IOI histogram, then phases using multiple agents             |
| Raphael (2001)               | MIDI, audio | Tactus, time quantization | Probabilistic generative model for onset times; MAP estimation (Viterbi)              |
| Cemgil & Kappen (2003)       | MIDI        | Tactus, time quantization | Probabilistic generative model for onset times; sequential Monte Carlo methods        |
| Goto & Mur-aoka (1995, 1997) | Audio       | Meter                     | Extract onset components; IOI histogram; multiple tracking agents                     |
| Scheirer (1998)              | Audio       | Tactus                    | Bank of comb filters to analyze periodicity of power envelopes at six subbands        |
| Laroche (2001)               | Audio       | Tactus, swing             | Extract discrete onsets; maximum-likelihood estimation                                |
| Sethares & Staley (2001)     | Audio       | Meter                     | Calculate RMS-energies at 1/3-octave subbands; apply a periodicity transform          |
| Gouyon et al. (2002)         | Audio       | Tatum                     | First find periods (IOI histogram), then phases by matching isochronous pattern       |
| Klapuri et al. (to appear)   | Audio       | Meter                     | Measure degree of accentuation; bank of comb filters; probabilistic model             |



80

## Aubio NoteOnset Detection

```
// run pvoc & rms onset detector
aubio_pvoc_do (pv, ibuf, fftgrain);
aubio_onsetdetection(o, fftgrain, onset);

// moving mean adaptive thresh. peak pick
isonset = aubio_peakpick_pimrt(onset, parms);

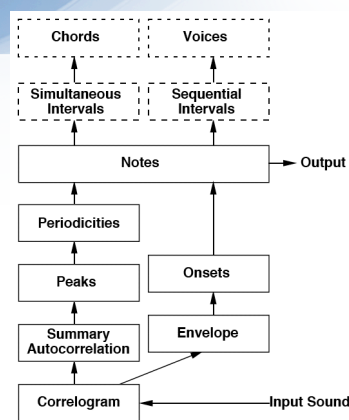
if (isonset) {
    // if at note onset
    if (aubio_silence_detection(ibuf, threshold2) == 1)
        isonset=0;
    else
        // play a sound on beat onset
        for (pos = 0; pos < overlap_size; pos++)
            obuf->data[0][pos] = woodblock->data[0][pos];
}
```



81

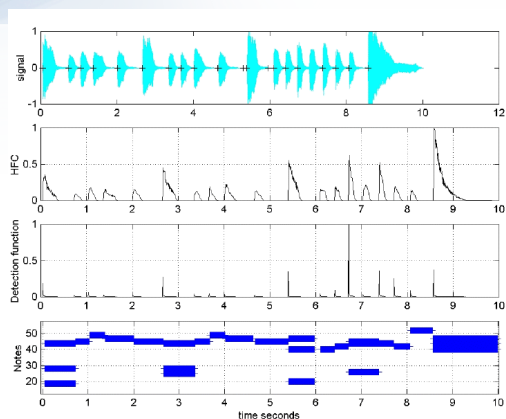
## Automatic Transcription

- Expected genre and its rules
- Multi-level peak continuation
- Many approaches possible



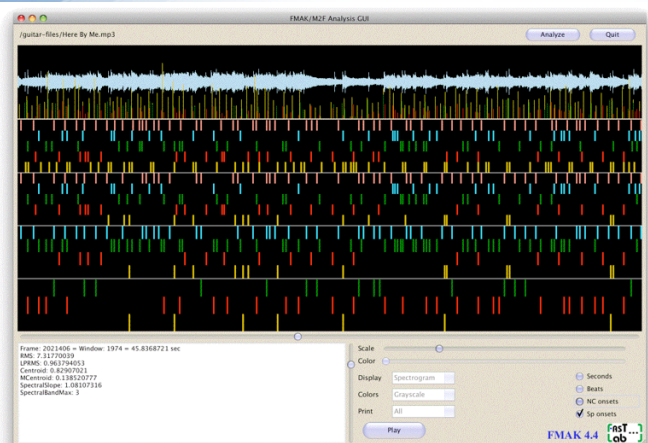
82

## Piano Transcription by Bello



83

## Music-2-Frets



84

## Other applications

- Source separation
- Similarity match
- Search (by various criteria)
- Play-list generation
- Summarization (thumb-nailing)
- Content-ID (finger-printing, tagging)



85

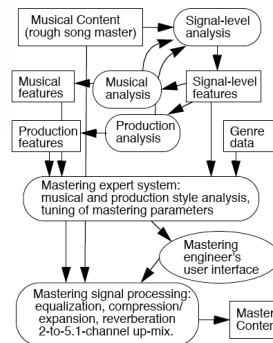
## Examples: MIREX Results

| MIREX Task Results                       |                         |          |                             |                         |          |                        |                         |          |                                     |                         |       |
|--|-------------------------|----------|-----------------------------|-------------------------|----------|------------------------|-------------------------|----------|-------------------------------------|-------------------------|-------|
| Audio Genre Classification (Generalized) |                         |          | Audio Artist Identification |                         |          | Multi-Pitch Estimation |                         |          | Audio Chord Detection (Pre-trained) |                         |       |
| Rank                                     | Participant             | Accuracy | Rank                        | Participant             | Accuracy | Rank                   | Participant             | Accuracy | Rank                                | Participant             | Score |
| 1  | University of Toronto   | 88.1%    | 1                           | Manuel R. Silva (1)     | 47.0%    | 1                      | NLS, Daniel & Cheng (2) | 0.685    | 1                                   | Media & Culture         | 0.68  |
| 2  | University of Jyväskylä | 86.1%    | 2                           | Manuel R. Silva (2)     | 47.0%    | 2                      | NLS, Daniel & Cheng (2) | 0.673    | 2                                   | Media & Culture         | 0.68  |
| 3  | Manuel R. Silva (1)     | 85.1%    | 3                           | Manuel R. Silva (2)     | 47.0%    | 3                      | Thomas & Riecke (2)     | 0.658    | 3                                   | Media & Culture         | 0.68  |
| 4  | Manuel R. Silva (2)     | 85.1%    | 4                           | University of Jyväskylä | 46.0%    | 4                      | Thomas & Riecke (2)     | 0.653    | 4                                   | Media & Culture         | 0.68  |
| 5  | Manuel R. Silva (1)     | 85.1%    | 5                           | University of Jyväskylä | 46.0%    | 5                      | Thomas & Riecke (2)     | 0.653    | 5                                   | Media & Culture         | 0.68  |
| 6  | University of Jyväskylä | 85.1%    | 6                           | University of Jyväskylä | 46.0%    | 6                      | Thomas & Riecke (2)     | 0.653    | 6                                   | Media & Culture         | 0.68  |
| 7  | University of Jyväskylä | 85.1%    | 7                           | University of Jyväskylä | 46.0%    | 7                      | Thomas & Riecke (2)     | 0.653    | 7                                   | Media & Culture         | 0.68  |
| 8  | University of Jyväskylä | 85.1%    | 8                           | University of Jyväskylä | 46.0%    | 8                      | Thomas & Riecke (2)     | 0.653    | 8                                   | Media & Culture         | 0.68  |
| 9  | University of Jyväskylä | 85.1%    | 9                           | University of Jyväskylä | 46.0%    | 9                      | Thomas & Riecke (2)     | 0.653    | 9                                   | Media & Culture         | 0.68  |
| 10                                       | University of Jyväskylä | 85.1%    | 10                          | University of Jyväskylä | 46.0%    | 10                     | Thomas & Riecke (2)     | 0.653    | 10                                  | Media & Culture         | 0.68  |
| 11                                       | University of Jyväskylä | 85.1%    | 11                          | University of Jyväskylä | 46.0%    | 11                     | Thomas & Riecke (2)     | 0.653    | 11                                  | Media & Culture         | 0.68  |
| 12                                       | University of Jyväskylä | 85.1%    | 12                          | University of Jyväskylä | 46.0%    | 12                     | Thomas & Riecke (2)     | 0.653    | 12                                  | Media & Culture         | 0.68  |
| Audio Chord Detection (Threat-free)      |                         |          |                             |                         |          |                        |                         |          |                                     |                         |       |
| Rank                                     | Participant             | Score    | Rank                        | Participant             | Score    | Rank                   | Participant             | Score    | Rank                                | Participant             | Score |
| 1  | University of Jyväskylä | 0.68     | 1                           | University of Jyväskylä | 0.68     | 1                      | University of Jyväskylä | 0.68     | 1                                   | University of Jyväskylä | 0.68  |
| 2  | University of Jyväskylä | 0.68     | 2                           | University of Jyväskylä | 0.68     | 2                      | University of Jyväskylä | 0.68     | 2                                   | University of Jyväskylä | 0.68  |
| 3  | University of Jyväskylä | 0.68     | 3                           | University of Jyväskylä | 0.68     | 3                      | University of Jyväskylä | 0.68     | 3                                   | University of Jyväskylä | 0.68  |
| 4  | University of Jyväskylä | 0.68     | 4                           | University of Jyväskylä | 0.68     | 4                      | University of Jyväskylä | 0.68     | 4                                   | University of Jyväskylä | 0.68  |
| 5  | University of Jyväskylä | 0.68     | 5                           | University of Jyväskylä | 0.68     | 5                      | University of Jyväskylä | 0.68     | 5                                   | University of Jyväskylä | 0.68  |
| 6  | University of Jyväskylä | 0.68     | 6                           | University of Jyväskylä | 0.68     | 6                      | University of Jyväskylä | 0.68     | 6                                   | University of Jyväskylä | 0.68  |
| 7  | University of Jyväskylä | 0.68     | 7                           | University of Jyväskylä | 0.68     | 7                      | University of Jyväskylä | 0.68     | 7                                   | University of Jyväskylä | 0.68  |
| 8  | University of Jyväskylä | 0.68     | 8                           | University of Jyväskylä | 0.68     | 8                      | University of Jyväskylä | 0.68     | 8                                   | University of Jyväskylä | 0.68  |
| 9  | University of Jyväskylä | 0.68     | 9                           | University of Jyväskylä | 0.68     | 9                      | University of Jyväskylä | 0.68     | 9                                   | University of Jyväskylä | 0.68  |
| 10                                       | University of Jyväskylä | 0.68     | 10                          | University of Jyväskylä | 0.68     | 10                     | University of Jyväskylä | 0.68     | 10                                  | University of Jyväskylä | 0.68  |
| 11                                       | University of Jyväskylä | 0.68     | 11                          | University of Jyväskylä | 0.68     | 11                     | University of Jyväskylä | 0.68     | 11                                  | University of Jyväskylä | 0.68  |
| 12                                       | University of Jyväskylä | 0.68     | 12                          | University of Jyväskylä | 0.68     | 12                     | University of Jyväskylä | 0.68     | 12                                  | University of Jyväskylä | 0.68  |

86

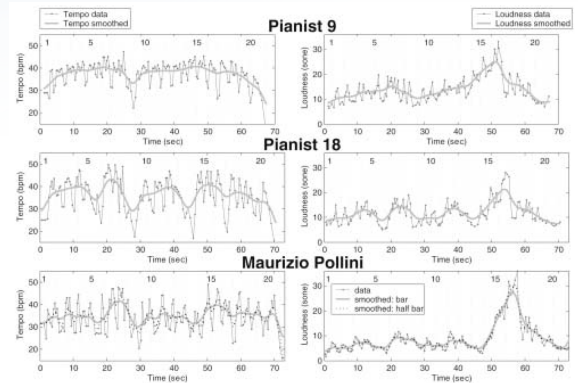
## In the Literature

- Accompaniment
- Transcription
- Segmentation
- Classification
- Instrument ID
- Mapping experts
- Fingerprinting



87

## Performance Analysis



88

## Performance Rule Extraction

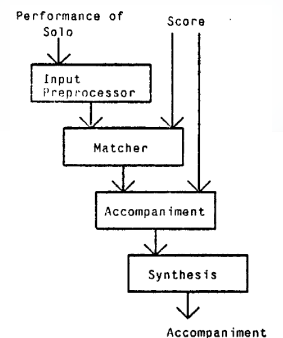
- Given a set of performance rules
  - Lerdaahl & Jackendoff
- Segment and analyze performance into tempo-phrasing functions
- Search for a weighting of the rules to match the performance
- Learn a performer's style



89

## Automatic Accompaniment

- Score format
- Input format
- Matching algorithm
- Output format
- Output generation
- Handling failures



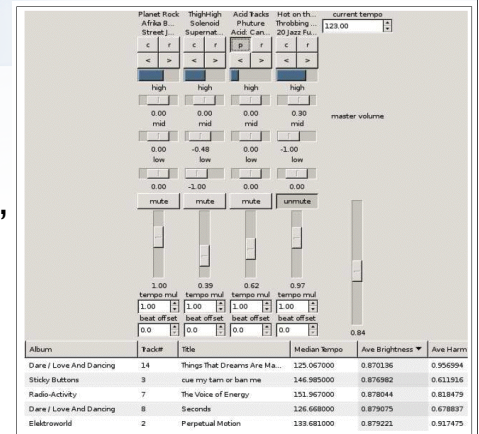
90

## Searching Musical Structures

- The Siren opera browser for Wozzeck

91

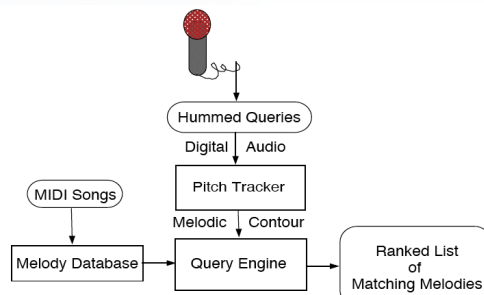
## DataJockey, A. Norman



92

## Query-by-Humming

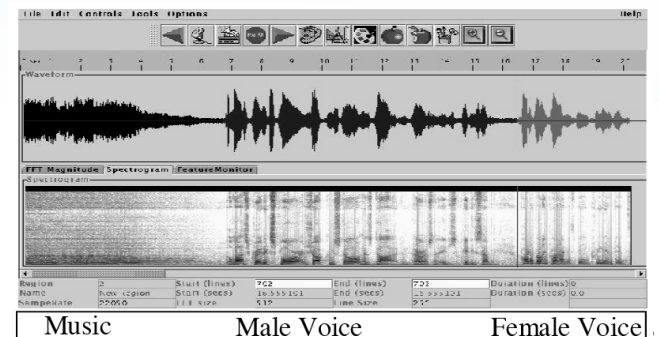
- Limited examples abound



93

## Content-Aware Tools

- What would you want a “smart editor” to do?



94

## Audio Summarization and Thumb-nailing

- Audio thumb-nailing assumes good segmentation, and location of the most typical short segment
  - Choose most-frequently repeated section (is that appropriate?)
  - Choose first incidence of common section
  - Some applications require a single note or a common signal window?

95

## Content Matching, Finger-printing

- Important for indexing and stream-following (e.g., segmentation of video)
- Genre classification (speech, music, effect) useful for segmentation and indexing
- Exact finger-printing for digital rights management (some systems in use)
- Still difficult to avoid false positives and not be easily spoofed

96

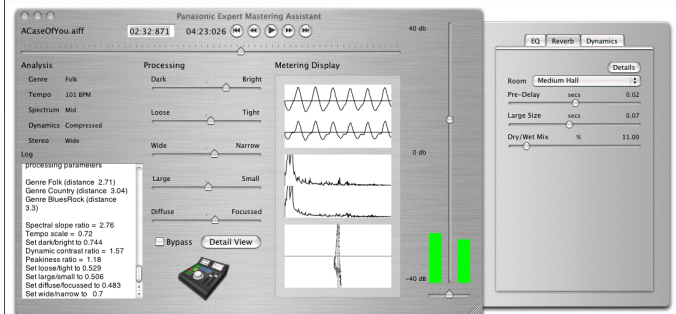
## Expert Mastering Assistant

- Analyze, classify a song
- Compare it to its “genre representative” taken from a carefully crafted DB
- Use expert rules to determine optimal processing parameters
- Perform DASP in real-time user feedback (display) and interaction



97

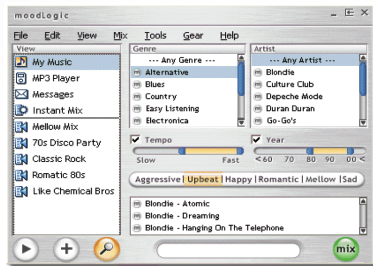
## EMA GUI



98

## Searching, Indexing, and Players

- The most common MDB application!
- Players often suffer from poor (or no) derived metadata or sophisticated indexing



- Some recent systems are changing this...
- There are many hard issues



99

## Applications

- Feature vector pruning, data reduction
- Segmentation examples: phonemes, musical segments
- SVMs for classification
- CURE for clustering
- Multipitch estimation, source separation, denoising



100

## Lab 3

- Continue Lab 1 in MATLAB
- Continue Lab 2 in C/C++
- Work on 2nd-stage processing using the SampleAnalyzer app (add a couple new features and fancier averaging, smoothing, thresholding, etc.)
- Prepare to do bulk feature extraction on the CAL500 data set to a CSV or MATLAB file



101