Query By Humming: A Survey

Eugene Weinstein
NYU and Google
eugenew@cs.nyu.edu

Introduction

- Searching for music is cool…
  - … but how to search for the tune stuck in your head?
  - Maybe type notes into your search engine?
- Humming is a natural way for humans to represent music
  - Music librarians/store clerks often act as query by humming engines

Problem Statement

- Given recording of hummed tune, identify song being hummed
- System needs to be robust to
  - Poor humming
    - Wrong pitch
    - Wrong note duration
    - Wrong key
  - Noise and distortion

Un-Motivation

- Pardo *et al*, '03: compare human to computer performance in recognizing hummed queries
- Bad news: Two humans with graduate degrees in music performance get <90% accuracy on *their own* recordings
- Average human accuracy: 66%

Table 1. Human Performance vs. Machine Performance

	Singer 1	Singer 2	Singer 3	Mean
Singer 1	96%	71%	79%	82%
Singer 2	50%	82%	46%	59%
Singer 3	71%	76%	89%	79%
Other2 Singers	61%	74%	63%	66%
String Matcher (Global)	29%	24%	39%	31%
String Matcher (Local)	36%	41%	71%	49%
HMM (Forward)	21%	35%	68%	41%
N	28	17	28	

One-slide Summary of Approaches

- Detect coarse melodic contour, retrieve by string search [Ghias *et al.*, 1995]
- Add rhythm information [McNab *et al.*, 1996]
- Use beat information [Chai *et al.*, 2002]
- Use HMMs to represent song database [Shiffrin *et al.*, 2002]
- Dynamic Time Warping (DTW) based algorithm, match waveform directly [Zhu *et al*., 2003]

Approach Classification: Features

- Features: almost all approaches try to detect pitch
  - Some add rhythm
  - Most eventually convert pitch to notes (or up-down sequences)
- Pitch detected by
  - Heuristics
  - Autocorrelation
  - Statistical methods (HMM)
- Some more recent approaches match directly to database of songs
  - Dynamic time warping (DTW)

Approach Classification: Retreival

- Matching recording to song database
  - Nearly all research uses MIDI files (note streams) as database
    - Formulate retrieval as matching task
  - Retrieval via
    - Approximate sequence matching algorithms [Ghias *et al.*, 1995; many others]
    - Statistical models (HMMs) [Shifrin *et al.*, 2002; Unal *et al.*, 2004]
    - Direct comparison to waveform using Dynamic Time Warping (DTW) [Zhu *et al*., 2003]

Ghias *et al*, 1995 (Cornell)

- Focused on accurately tracking humming pitch, by
  1. Autocorrelation – detecting audio amplitude peak frequency [Rabiner *et al*., 1976]
     - Problem: aliasing, slow
  2. Maximum Likelihood pitch tracking [Wise *et al*., 1976]
     - Problem: way too slow
  3. Cepstrum analysis [Oppenheim 1969]
     - Problem: not very accurate for humming

Ghias *et al*: String Matching

- Pitch transitions encoded as melodic contour: S=same note, U=up, D=down
  - E.g., Beethoven's 5th: – S S D U S S D
  - This is known as Parsons code [Pasons, 1975]
- Use approximate string matching algorithm [R. Baeza-Yates *et al.*, 1992]
  - Find instances of pattern P in string T with at most k mismatches
  - If n=length(T) and m=length(P), Σ=size of alphabet, average-case runtime is

$$O\left(n\left(1+\frac{m}{\Sigma}\right)\right)$$

Ghias *et al.*: Evaluation

- Database: 183 MIDI songs
  - Use “a number of” heuristics to extract melody from MIDI tracks
- A number of humming recordings (don’t say how many!)
  - All male voices
- Index 10-12 note n-grams
  - Sufficient to “discriminate 90% of the songs”
- Close to 100% accuracy under “ideal conditions”
  - Pause between each note pair, hit each note strongly

McNab, *et al.* 1996 (U. Waikato, NZ)

- Query by humming system called MELDEX
- Track pitch by detecting periodicity in time domain [Gold & Rabiner, 1969]
- Add rhythm information by analyzing note duration
- Use pitch, rhythm, up-down contour (like Ghias)
- Only match the beginning of the song
- Retrieval by "dynamic programming" algorithm capable of exact or approximate matching.

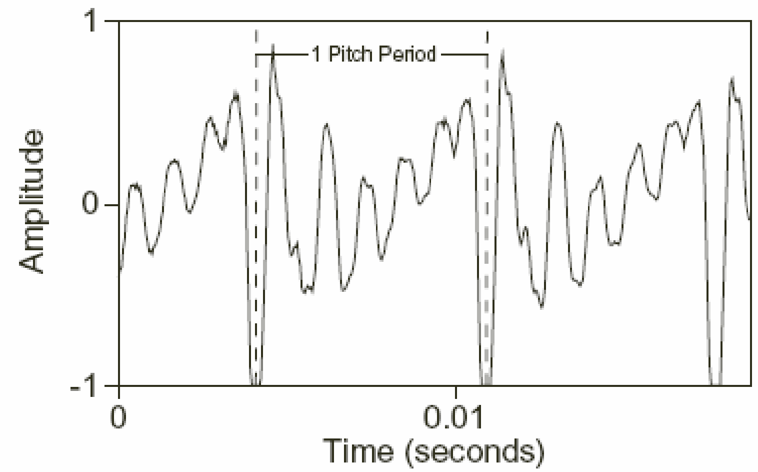


Figure 1. Acoustic waveform of *ah*

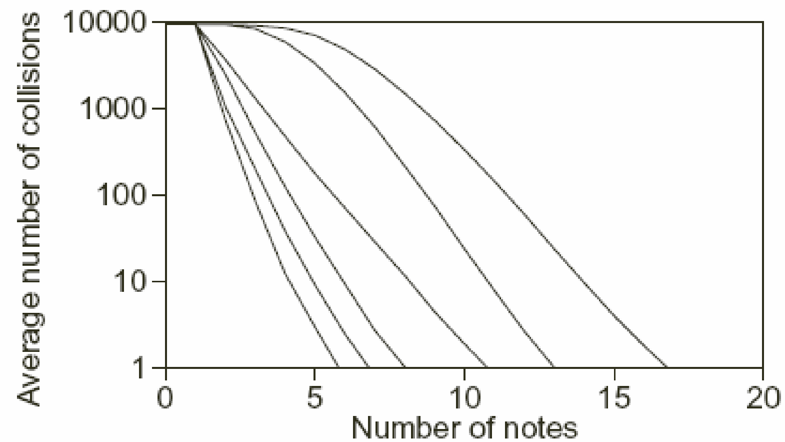
McNab: Performance Experiment

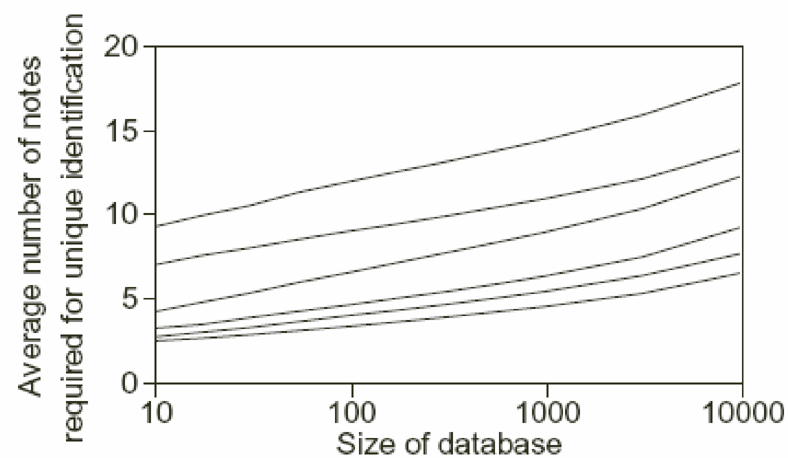
- Ten subjects
  - Six with extensive performance experience
  - Four with little or no musical education
- Ten folk songs
- Conclusion: people have trouble with
  - Changes in key
  - Large differences between adjacent notes
- Easiest melodies: stepwise changes in small increments
- Accuracy best in people with amateur performance experience
  - Not much correlation with formal musical education

Song	Number of singers	Ending "in key"
Bridge Over Troubled Water	7	5
Hound Dog	8	8
King of the Road	8	3
Memory	10	8
Moon River	10	4
Pokare kare ana	10	7
Puff, The Magic Dragon	10	8
Summertime	9	4
Yankee Doodle	10	9
Yesterday	9	3

McNab *et al*: More experiments

- Experiment 1: With 9,600 songs, how many notes are needed to uniquely identify a song?
- Experiment 2: How does the number of notes needed vary with the database size?
- Line index (left to right)
  - exact interval and rhythm
  - exact contour and rhythm
  - exact interval
  - exact contour
  - approximate interval and rhythm
  - approximate contour and rhythm





Prechelt *et al.* 2001 (U. Karlsruhe)

- Tuneserver system: Query by whistling
  - Gender-independent
  - Much lower frequency range than humming or singing
- Approach: convert pitch transitions to U/D/S contour, as in Ghias, *et al*
- Identify pitch simply by detecting maximum-energy frequency
  - Works because whistling should contain only dominant frequency and overtones
- Match against song database by finding song with minimum edit distance from recording
  - Insertion/deletion/substitution weights trained to provide maximum empirical discrimination

Prechelt *et al*: Experiments

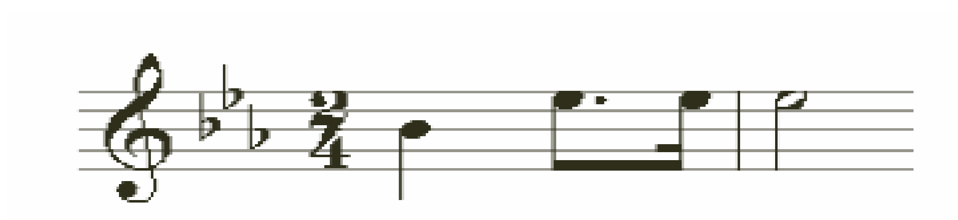
- Database: 10,370 classical music themes published in [Parsons, 1975]
- 24 subjects
  - 18 computer scientists, a few musicians
- Recordings made with laptop microphone
- 106 recordings
  - Two required songs, and two songs of the subject's choosing

Prechelt *et al*: Results

- Accuracy figures:
  - 77% of queries: correct song in top 40
  - 44% of queries: correct song is top match
  - 81% / 47% if you adjust for songs hummed so poorly that even the accurate U/D/S sequence is incorrect
- Most inaccuracies due to breathing
  - Recordings with no breathing: 86% / 59%

Chai *et al*, 2002 (MIT)

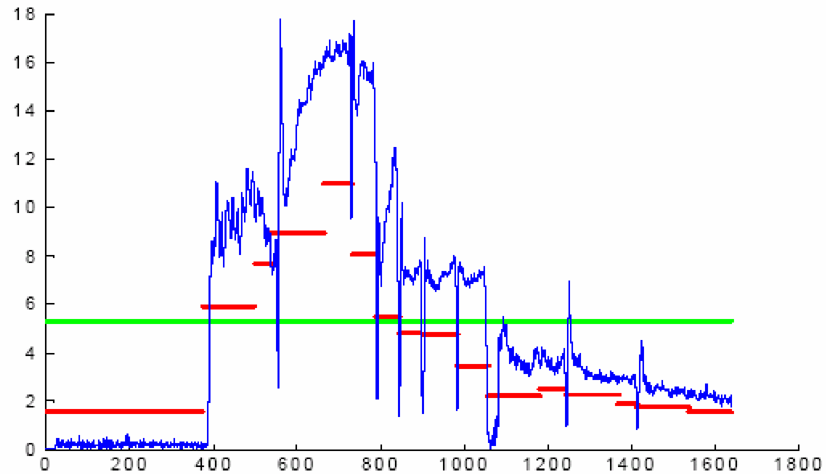
- Compute rough melodic contour
  - U/D/S – but with five contour levels
- Algorithm: count number of equivalent transitions in each beat
  - Difference from previous work: take into account the beat structure of the songs
  - However, no rhythm information is used



“Here Comes the Bride”
<TimeSig, Contour, Beat #> =
<[2 4], [* 2 0 0], [1 2 2 3]>

Chai *et al*: Signal Processing

- Notes detected by amplitude-based note segmentation
  - Use amplitude thresholds to detect voicing onset, offset
- Pitch tracking by autocorrelation
- Beat information obtained by user input
  - Option 1: user inputs desired beat, hums to drum track
  - Option 2: user clicks mouse at each beat



Chai *et al*: Experiments

- Experimental setup:
  - Database of 8,000 MIDI songs
  - 5 test subjects
    - Some with, some without musical training
  - Each subject asked to hum 5-13 songs
    - 45 total recordings
- Compare
  - Two new algorithms (consider beat information)
  - Edit distance type algorithm for pitch only
- Subjects with musical training do better!
- Beat information helps (but interface is not that natural)

	New Algo 1	**New Algo 2**	**ED type**
Top match	53%	46%	44%
Top 10	64%	51%	56%

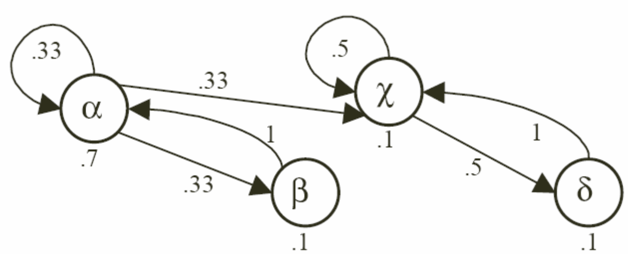
Shiffrin *et al*, 2002 (U. Mich.)

- Subjects hum syllables (e.g., “la la la”)
- Segment audio into 10ms frames
- Resolve each frame to pitch level using pitch tracker [Tolonen ’00]
- Regions of pitch stability: notes
- Feature vector: [Δpitch, Δtime]
- Hummed song identified by HMMs

Shiffrin *et al*: HMM Representation

- States are note transitions
  - Unique state for each [Δpitch, Δtime] tuple
- Traversing an arc represents moving between notes
- State, transition weights set according to counts in MIDI database of in-set songs
- Retrieval by HMM “forward” algorithm [Rabiner ’89]
  - No search

Delta pitch	2	2	1	2	-2	-1	-2	-2
IOI	1	1	1	1	1	1	1	1
IOI ratio	1	1	1	1	1	1	1	1
State	α	α	β	α	χ	δ	χ	χ



Shiffrin *et al*: Experiments

- 277 MIDI songs in a variety of genres
- Extract 2,653 monophonic themes [Meek 2001]
- Subjects hum any “significant” part of the song
  - Hum six in-set songs each
  - Four subjects, two with grad degrees in music performance (24 test recordings total)
- Match against all themes by HMM forward, edit distance

Table 1: Number of cases by rank of correct answer

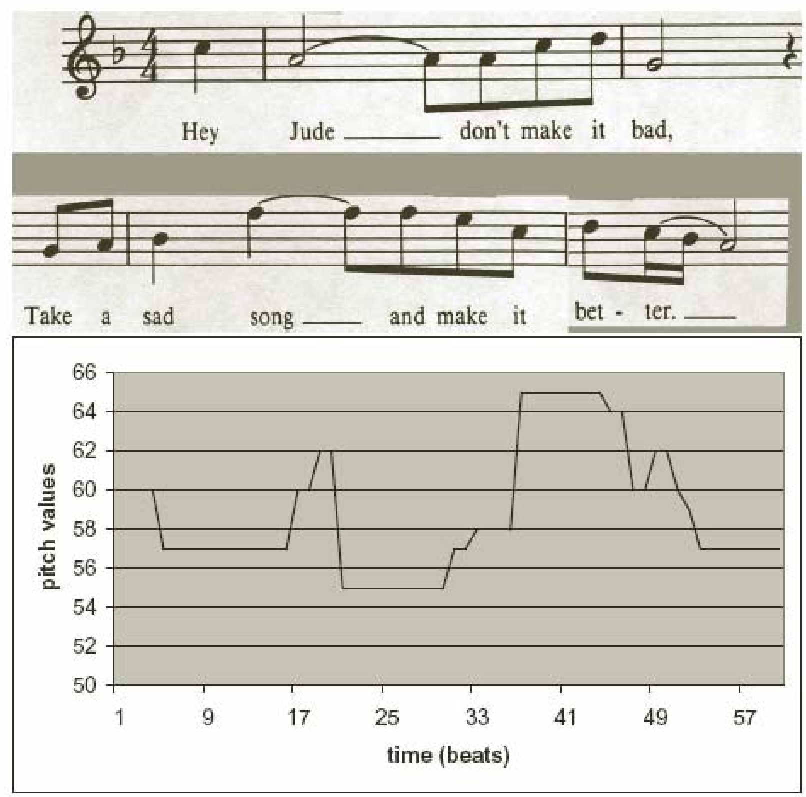
System	HMM		String Matcher	
Rank of Correct Answer	Number of Cases	Cumulative Percentage	Number of Cases	Cumulative Percentage
1	10	41.7%	4	16.7%
2 to 5	4	58.3%	1	20.8%
6 to 10	0	58.3%	1	25.0%
11 to 25	3	70.8%	2	33.3%
26 to 50	1	75.0%	4	50.0%
51 to100	3	87.5%	4	66.7%
Over 100	3	100.0%	8	100.0%

Zhu *et al*, 2003 (NYU)

- Problem #1: melodic contour approaches flawed
  - It's hard to detect notes in hummed tune
  - Contour does not identify a song uniquely
    - E.g., 330/2,697 tracks contain same six-note contour [Uitdenbogerd, 1998]
- Problem #2: people can't hum
  - Thus, cannot refine contour for better precision
  - Forcing people to hum with syllables (e.g., "da da da") is unnatural
- Proposal: treat hummed query as time series
  - Match audio directly against reference recording
  - No note detection

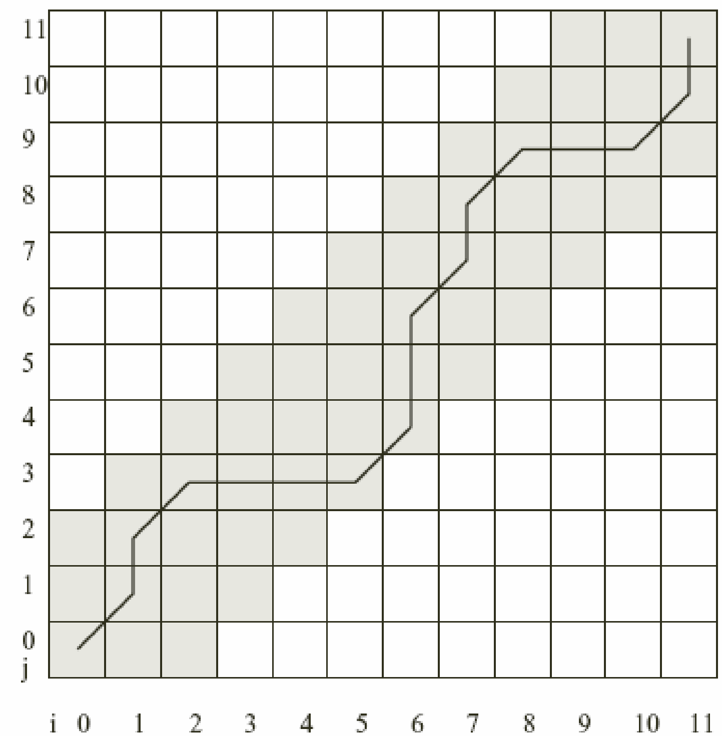
Zhu *et al*: Approach

- Treat reference and hummed melodies as time series
  - Segment audio into 10ms frames
  - Resolve each frame to pitch level using pitch tracker [Tolonen '00]
  - No note segmentation
- Match entire song sub-sequences (i.e., no partial tune matching)



Zhu *et al*: Time Series Retrieval

- Global tempo may be off by ±50%
  - Apply uniform time warping (UTW)
  - Basically, stretches or compresses recording
- But still might have local tempo variations
  - Apply local dynamic time warping (LDTW)
- Novel combination of UTW and LDTW



Zhu *et al*: DTW/UTW Overview

- Given sequences x: $x_1 \cdots x_n$, y: $y_1 \cdots y_m$
  - Let $x_{rest}=x_2 \cdots x_n$, $y_{rest}=y_2 \cdots y_n$
  - Then DTW distance between x and y is:

$$D^2_{DTW}(x,y) = D^2(x_1,y_1) + \min \begin{cases} D^2_{DTW}(x, y_{rest}) \\ D^2_{DTW}(x_{rest}, y) \\ D^2_{DTW}(x_{rest}, y_{rest}) \end{cases}$$

  - LDTW: Just limit the range of x_{rest}, y_{rest}
  - UTW distance is:

$$D^2_{UTW}(x,y) = \frac{\sum_{i=1}^{mn} (x_{\lceil i/m \rceil} - y_{\lceil i/n \rceil})^2}{mn}$$

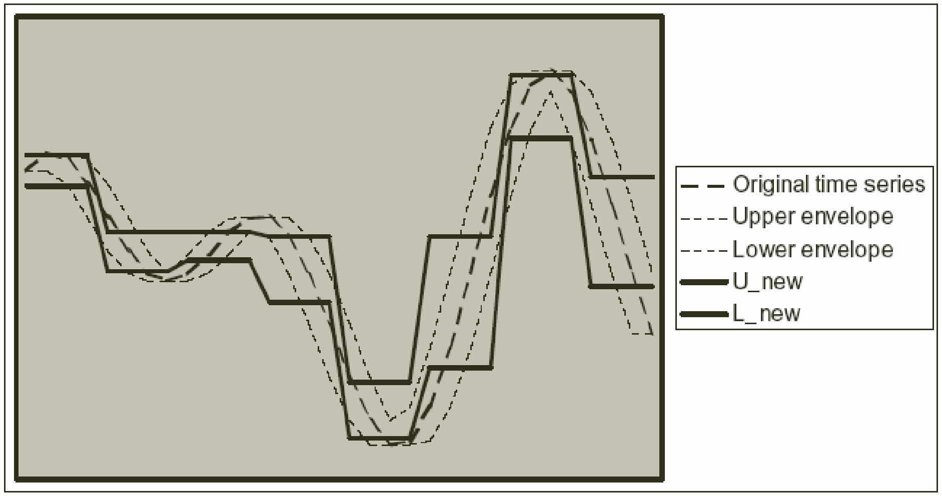
- Algorithm: do global UTW and local LDTW

Zhu *et al*: Contour Envelope

- In practice, DTW is costly to compute
- Also, want to reduce signal dimensionality for ease of indexing
- Solution: approximate DTW by computing "envelope" around pitch contour
  - Define k-envelope upper and lower bounds

$$
\begin{aligned}
x_i^L &= \min_{-k \le j \le k} (x_{i+j}) \\
x_i^H &= \max_{-k \le j \le k} (x_{i+j})
\end{aligned}
$$

  - Use novel piecewise aggregate approximation (PAA) variant (see paper)



Zhu *et al*: Finally the algorithm!

- Build an index structure (e.g., R* tree) containing all songs
- For a test recording:
    1. Compute envelope and PAA-type approximation
    2. Make ε-range query on index structure, get back list of candidates
    3. Pick candidate with smallest DTW distance to test recording

Zhu *et al*: Experiments

- Fifty Beatles songs
  - Segment into 1,000 15-30 note melodies
  - Collect a number of humming recordings
  - Pick 20 melodies by “better singers”
- Compare time series approach vs. standard contour matching approaches
- Only 4/20 recordings of poor singers matched perfectly

Rank	Time Series Approach	Contour Approach
1	16	2
2-3	2	0
4-5	2	0
6-10	0	4
10	0	14

Unal *et al*, 2004 (USC)

- Use HMMs to segment recording into notes
  - HMM trained on actual humming data
  - Standard speech setup (GMM acoustic model, Baum-Welch training, Viterbi decoding)
- Then, detect pitch by autocorrelation
- Features:
  - Pitch change contour
  - Duration change contour

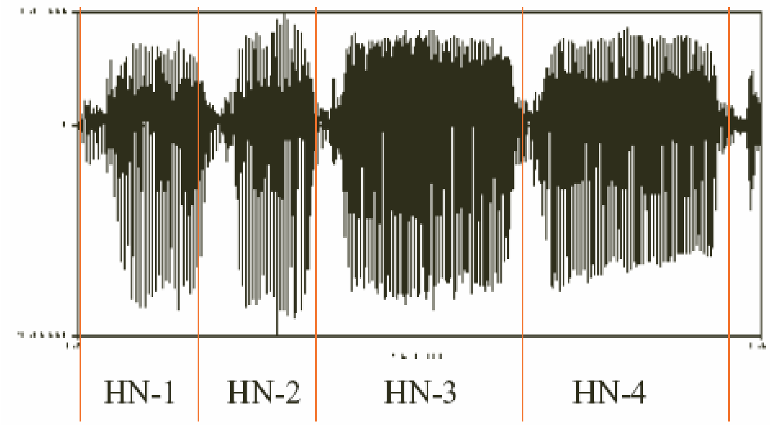


Figure 4. Segmented Notes and Labeling

Table 1. Pitch and Duration Transcription

	HN#1	HN#2	HN#3	HN#4
Pitch (Hz)	111.7	123.25	141.98	128.63
Duration (sec)	0.326	0.255	0.456	0.520
Pitch Transcription(*PT*)	0	1.703	2.449	-1.709
Duration Transcription(*DT*)	1	0.78	1.14	0.94

Unal *et al*: Indexing and Retrieval

- Identify regions of large and small pitch and duration change
- Fingerprint: two samples around landmark
- Compute similarity score
  - Difference between features of reference and test
- Rank results by similarity score

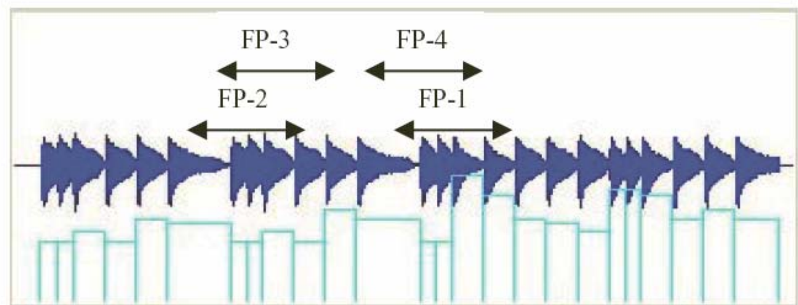


Table 2.1. FP1: largest pitch transition

PT	-4.62	-0.33	10.99	-2.21	-3.67
DT	0.48	0.86	2.42	0.94	1.06

Table 2.2. FP2: smallest pitch transition

PT	-0.81	-4.79	0.07	2.44	-1.92
DT	1.05	0.56	0.77	2.86	1.18

Table 2.3. FP3: largest duration change

PT	-4.79	0.07	2.44	-1.92	6.54
DT	0.56	0.77	2.86	1.18	0.91

Table 2.4. FP4: smallest duration change

PT	-1.92	6.54	-1.82	-4.62	-0.35
DT	1.18	0.91	1.01	0.48	0.86

Unal *et al*: Experiments

- Database: 200 MIDI files
- Test data: 250 humming pieces
  - Evenly split between trained, non-trained subjects

Table 4.1 Results for Non-Trained Subjects

Size of Database	50		100		200	
	Top of the list	Within first 5	Top of the list	Within first 5	Top of the list	Within first 5
K=1	%54	%92	%42	%86	%38	%80
K=2	%84	%100	%78	%90	**%72**	%88
K=3	%82	%100	%80	%86	%72	%86

Table 4.2 Results for Trained Subjects

Size of Database	50		100		200	
	Top of the list	Within first 5	Top of the list	Within first 5	Top of the list	Within first 5
K=1	%82	%96	%80	%94	%76	%88
K=2	%100	%100	%98	%100	**%94**	%100
K=3	%100	%100	%98	%100	%98	%100

Data Sets

- Only one publicly available corpus from USC [Unal *et al*., 2003]
  - License is still being worked out
- Several small corpora collected for experiments…
  - … but there are confidentiality issues
  - MIT Corpus [Chai *et al.*, 2002] not available
  - NYU corpus [Zhu *et al.*, 2003] available, but missing metadata

Summary

- Fewer than ten query by humming systems have been published
- Accuracy okay in favorable conditions
  - But, rigorous evaluation is scarce
- Some interesting approaches, but insights are not tremendous
- For us, two big questions:
  - Can we do better?
  - Is there a good application for this technology?

References

- Humming-specific
  - A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith. “Query by humming - musical information retrieval in an audio database,” In ACM Multimedia 95, 1995
  - R.J. McNab, L.A. Smith, I.H. Witten, C.L. Henderson and S.J. Cunningham, “Toward the digital music library: tune retrieval from acoustic input,” Proc. ACM Digital Libraries, 1996, pp 11--18.
  - L. Prechelt and R. Typke, “An interface for melody input,” ACM Trans. On Computer Human Interaction, 8, 2001.
  - W. Chai and B. Vercoe, “Melody Retrieval On The Web,” Proceedings of ACM/SPIE Conference on Multimedia Computing and Networking, Jan. 2002.
  - J. Shifrin, B. Pardo, and W. Birmingham. “HMM-Based Musical Query Retrieval,” in Joint Conference on Digital Libraries. 2002. Portland, Oregon
  - B. Pardo and W. Birmingham, “Query by Humming: How good can it get?,” Workshop on Music Information Retrieval, SIGIR 2003, Toronto, Canada, July 28 - August 1, 2003.
  - E. Unal, S. S. Narayanan, H.-H. Shih, Elaine Chew, C.-C. J Kuo, “Creating Data Resources for Designing User-centric Front-ends for Query by Humming Systems,” in 2003 Multimedia Information Retrieval (ACM-MIR03), November 2003.
  - Y. Zhu and D. Shasha, “Warping Indexes with Envelope Transforms for Query by Humming,” ACM SIGMOD 2003 International Conference on Management of Data, June 9-12, 2003, San Diego, California.
  - E. Unal, S. S. Narayanan, and E. Chew, “A Statistical Approach to Retrieval under User-dependent Uncertainty in Query-by-Humming Systems,” in 2004 Multimedia Information Retrieval (ACM-MIR04), October 2004.
- Non-humming-specific
  - A. V. Oppenheim, “A speech analysis-synthesis system based on homomorphic filtering” J. Acoustical Society of America, 45:458-465, February 1969.
  - D. Parsons. The Directory of Tunes and Musical Themes. Spencer Brown, Cambridge, 1975.
  - L.R. Rabiner J.J. Dubnowski and R.W. Schafer. Realtime digital hardware pitch detector. IEEE Transactions on Acoustics, Speech and Signal Processing, ASSP- 24(1):2–8, Feb 1976.
  - J. D. Wise, J. R. Caprio, and T. W. Parks, “Maximum likelihood pitch estimation,” IEEE Trans. Acoustics, Speech, Signal Processing, 24(5):418-423, October 1976.
  - R. A. Baeza-Yates and C. H. Perleberg, “Fast and practical approximate string matching,” Combinatorial Pattern Matching, Third Annual Symposium, pages 185-192, 1992.
  - Mongeau, M. and Sankoff, D. (1990) “Comparison of musical sequences.” Computers and the Humanities 24: 161–175.
  - Meek, C., Birmingham, W. Thematic Extractor, in Proceedings of ISMIR 2001 (Bloomington, IN, October 2001), 119-128.
  - Rabiner, L. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE. Vol. 77, No. 2, 1989, 257-286.