# A Dictionary Based Approach for Robust and Syllable-Independent Audio Input Transcription for Query by Humming Systems

Erdem Unal Shrikanth Narayanan Elaine Chew Panayiotis G. Georgiou Nathan Dahlin
Integrated Media Systems Center
Viterbi School of Engineering, University of Southern California, CA
unal@usc.edu shri@sipi.usc.edu echew@usc.edu georgiou@sipi.usc.edu dahlin@usc.edu

## ABSTRACT

Transcription from audio to musical representation is a challenging problem for Query by Humming (QBH) systems. In this paper, we propose a two step note transcription process consisting of an algorithm that uses a speech recognizer for note segmentation followed by signal processing for robust location and capture of pitch and duration in the humming audio input. In contrast to most Hidden Markov Model based approaches to QBH systems that model and classify humming into a single universal model, we designed a flexible speech recognizer that allows different types of humming sounds in the input for providing efficient and accurate note segmentation and transcription. We use novel statistical energy and pitch analyses to correct potential insertion and deletion errors to augment the system's performance, and evaluate our algorithm with precision and recall tests. Using a large database previously amassed, we test various system configurations, providing results for note segmentation with and without the proposed augmentations. The augmented system robustly recognizes the location of humming notes with a precision and recall *F* measure of 0.84. As a second validation, we use the results of the transcription system in melody retrieval and found, for a database of 1000 melodies, a 76% retrieval accuracy with automatically extracted queries, and a 83% retrieval performance with manually transcribed queries. Sensitivity analysis shows that, while it is possible to locate the position of the hummed notes accurately, incorrect segmentation results can have a negative effect in the retrieval performance of the QBH system.

## Categories and Subject Descriptors

H.5.5 [**Information Interfaces and Presentation]:** Sound and Music Computing – Methodologies and techniques; Signal analysis, synthesis, and processing

*AMCMM'06*, October 27, 2006, Santa Barbara, California, USA.
Copyright 2006 ACM 1-59593-501-0/06/0010...$5.00.

## General Terms

Measurement, Performance, Experimentation, Human Factors,

## Keywords

Query by Humming, HMM based transcription, Retrieval

## 1. INTRODUCTION

Music Information Retrieval (MIR) is rapidly gaining attention among researchers in a wide range of areas including signal processing, media technologies, databases, human factors, and interactive music application. Technology progress, such as that in storage capacities of web servers, makes MIR even more attractive, since the Internet is one of the main resources for multimedia data mining today. Various kinds of audio retrieval technologies have been implemented for easy access to the desired music data [16]. Query-by Humming (QBH) is one of these ongoing MIR research technologies, which is becoming increasingly popular and more sophisticated. The main goal of QBH systems is to take human humming audio as input, and use it as a query to retrieve music from a database as accurately as possible.

In this paper, the problem of melodic transcription and representation in the front end of QBH Systems is discussed. Each humming unit represents a single note in a melody. One of the main goals of the front-end system is to provide correct segmentation of the humming notes one from another. After segmentation, the pitch and duration values for each segmented note can then be extracted. We propose a two-stage note transcription process comprising of first, an algorithm that uses a speech recognizer for note segmentation and, then signal processing techniques for precisely extracting pitch and duration information. This transcription process allows for multiple humming syllable types in various forms, which is typical of real world humming data, and incorporates post-processing using pitch and energy features to enhance system performance.

Here, we provide justification for the use of a speech recognizer for note segmentation. Humming can be defined as the reproduction of a melody without the words. Instead of the lyrics, fixed syllables such as /Da/, /Na/, /Ta/, /Ra/ and their different pronunciations are used. The main difference in salient feature sets between spontaneous speech syllables and humming units is the pitch and duration structure. The duration of the notes in the humming sequence (depending mainly on the tempo of the humming) is more likely to be longer than the syllables used in spontaneous speech. On the other hand, since each humming unit represents a music note, the pitch in the hummed notes tends to be more stable, while pitch in spontaneous speech may vary with expressiveness. Since the duration and the pitch contour characteristics are the only differences, and these differences do not significantly affect speech recognition performance, humming can be considered ”meaningless” (non-lexical) speech for the purposes of humming unit (note) segmentation. This suggests that adopting an automatic speech recognizer for locating hummed notes in the input query could be a feasible approach for segmentation. Admittedly, there exist alternative methods to using a speech recognizer to detect note onsets, such as through analysis of the short-term energy profile. But these methods are not robust as standalone techniques.

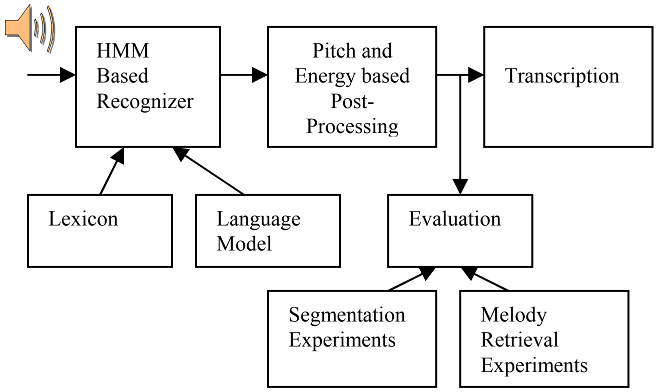

**Figure 1. Proposed Note Segmentation Algorithm**

Figure 1 shows a diagram of our proposed note transcription system. In order to perform correct note segmentation, we design a robust Hidden Markov Model (HMM) based speech recognizer. The speech recognizer’s task is to perform pre-segmentation by using statistical time-series note models. In the recognizer, we create 4 general models for different types of humming syllables that can be expected in humming audio. Our lexicon (dictionary used in the recognizer) groups the humming syllables into main categories with respect to their linguistic structures for accurate recognition. We train the generic humming models with available speech data corpora (see Section 2), and a “language” (note sequence) model with transcribed real world humming data. The output of a phoneme-level speech recognizer is post processed with energy and pitch features to detect the humming note boundaries. It is then segmented into audio chunks that correspond to the detected humming notes.

Finally, we evaluate the segmentation performance of the system with precision and recall (*F measure*) analysis. We also use the output of the recognizer for melody retrieval experiments in order to investigate how segmentation errors impact the performance of QBH systems.

## 1.1 Related Work

In this section, we describe the melodic transcription and representation schemes adopted in previous QBH systems. Only very few articles provide details on note transcription techniques. Hence a large part of the literature review focuses on the melodic representation schemes employed in these QBH MIR systems. The transcription method proposed in this paper can be used as a front-end module for many of the QBH systems invoked in this section.

Ghias et al. [1] has been credited as one of the first to propose a QBH system that converts audio into melodic contour representation. The contour representation has been used widely by many QBH systems that followed. Ghias used autocorrelation for pitch tracking in order to encode the audio into a 3 level alphabet of pitch contours (U, D and S). McNab et al. [2], [3] introduced the duration concept into the melodic representation scheme. Tree based search techniques have been used by Roland et al. [4] and Blackburn et al. [5] as an improvement to McNab’s system, in order to get better and more efficient retrieval results. Jang et al. [6] expanded the 3 level alphabets to one using the semitone (half step) distance measure. Kosugi et al. [9] applied Euclidian Distance search in a system in which the input and database elements are segmented into fixed length windows. In later studies, Kosugi et al. [10] tested the fixed length transcription technique and the Euclidian Distance, using tone transition and tone distribution features. Hu et al. [7] also used fixed sized frames in melody transcription, noting that windowing handles the errors of transcription, transposition invariance, overall tempo difference, and local tempo variation.

Shih et al. [11] used HMM’s for note segmentation in the front-end of a QBH system. Clarisse et al. [8] first evaluated existing transcription systems, such as those by Meldex, Pollastri, Autoscore, etc. Observing that these systems are not adequate for human level performance, Clarisse et al. constructed an Auditory Model based transcription system for the front-end of their QBH systems.

In Cuby-Hum, Pauws [12] designed a new transcription technique that processes the input in terms of energy and pitch to detect note onsets and locations, and quantizes the input into semitone representations. Pauws then used Dynamic Programming to align query and database elements in the retrieval side of the system.

Pardo et al [13] tried two different similarity measurement techniques for the hummed queries. The first approach estimated the distance between the target melodies and the database entries using an edit measure. In the second approach, target melodies in the database were represented as HMM’s, and the input is represented as observation sequences. A target is judged similar to the query if the HMM representation likely produces the query.

As a supporting study to QBH system design, we collected a large database of humming samples from 100 different people and statistically analyzed and categorized the uncertainty present in the human production of humming [14]. We also introduced a new retrieval technique that extracts finger prints from indexed audio input for efficient search in QBH systems in the case of perfect transcription [15]. Various system solutions described above are summarized in Table 1.

**Table 1. Summary of milestone QBH systems (Symbols describe pitch contour properties: U- up, D-down, S-same)**

| Authors | Representation | Search and Retrieval |
|---|---|---|
| Ghias | Pitch contour(U,D,S) | String Matching |
| McNab | Pitch contour(U,D,S) and duration | String Matching |
| Roland and Blacburn | Pitch contour(U,D,S) | Tree Based Search |
| Jang | Pitch contour in semi-tones and duration | Dynamic Programming |
| Kosugi, Hu amd Zhu | Fixed length window pitch info | Dynamic Programming |
| Shih | HMM contour ratios based pitch and duration | Tree based search |
| Pauws | Midi Representation | DTW, Edit Distance |
| Pardo | Midi Representation, HMM's | Edit Cost, Likelihood |

As seen from the Table 1, all proposed systems use a specific way to represent the audio input signal. Psychoacoustic research suggests that music is mentally perceived as relative pitch and duration contours of sound sequences. Not only does the ability for music production by humans, but also their ability for melody identification, is highly influenced by the way they perceive music [20]. Hence, as a first step, a robust way for extracting the pitch and duration of humming notes from the audio input needs to be implemented. This, in turn, means that the humming notes in the audio input have to be segmented as accurately as possible. A problem that pervades all QBH systems mentioned above is how one can accurately transcribe the audio input into the desired melodic representation. Since our proposed transcription system is designed to be robust against variability in humming, it can be used by most of the QBH systems as a front-end module for performing accurate audio to symbol transcription.

The remainder of the paper is organized as follows: Section 2 describes how the HMM-based approach is adopted from speech recognition and adapted for use in the segmentation of humming notes. Section 3 describes the segmentation performance evaluation in our experiments. Section 4 reports on comparisons of the retrieval performance of the proposed automatic note segmentation algorithm against that of expert human transcription. Section 5 concludes with some discussion and possible future directions.

# 2. FRONT END HUMMING RECOGNIZER

We invoke speech recognition technology for extracting note boundaries from a hummed query, followed by post-processing to clean up the speech recognizer's output and determine pitch. Speech recognition technology has been employed in a variety of applications; a detailed review can be found in [21].

This section presents details on the design of the humming recognizer system. Section 2.1 describes the adaptation of a speech recognizer for our purposes of note segmentation in the QBH context, and Section 2.2 our techniques for post processing the speech recognizer output to handle insertion and deletion errors.

## 2.1 Speech Recognizer

We developed a phoneme level speech recognizer using the Sonic [17] speech recognition system. Phonemes are mapped to syllables that represent humming notes. The syllable set was limited to those frequently encountered in real-world humming data. The statistical properties of a hummed note vary with the syllable used to sing the note. For this reason, statistical HMM models are selected to represent the different hummed notes and compensate for this variety. Instead of trying to estimate optimal parameters for a hand-built model for direct onset detection from energy and pitch, statistical approach is mostly preferred.

Mel frequency cepstral coefficients (MFFC 's) are the feature set for the statistical models, which are commonly used in speech recognition [21]. Four generic syllable model types, denoted by /Da/, /Ta/, /Na/, /Ra/, were defined in the lexicon, where each model represents a single type of consonant that is expected at the beginning of a hummed note (/Da/: voiced stops (*b, c, d, g*, etc…), /Ta/: unvoiced stops (*p, t, k*, etc…), /Na/: nasals (*m, n*), and /Ra/: liquids (*l, r*)).The lexicon also includes the different types of vowels (AA, AH, IH, AE) that are expected to follow the consonant to form the hummed syllable. The 4 generic syllable models provided in the dictionary aim at allowing all different types of syllables that can be used by the human subjects regardless of the consonant at the beginning of the humming syllable. The lexicon we used is summarized Figure 2.

| Syllable | Consonant | Phoneme | |
|---|---|---|---|
| DA | D | IX | (Voiced Stops) |
| | D | AE | |
| | D | IH | |
| | … | .... | |
| TA | T | IX | (Unvoiced Stops) |
| | T | AE | |
| | … | … | |
| RA | R | IX | (Liquids) |
| | R | AE | |
| | ... | ... | |
| | L | IX | |
| | L | AE | |
| | ... | ... | |
| NA | N | IX | (Nasals) |
| | N | AE | |
| | N | IH | |

**Figure 2. The syllable lexicon used by the automatic segmentation**

To obtain a good estimation of the statistical properties of the vocal sounds, considerable amounts of training data are needed. To bootstrap the models, the training was done using large corpora of read speech data (TIMIT, Wall Street Journal) that provided adequate coverage for the various phonemes. A bi-gram note sequence model was developed using the CMU Language Modeling Toolkit [18], from a set of transcribed humming data. Such a "language" model derived from real data transcription is believed to help in the cases, where different combinations of the note models may represent frequently occurring rhythmic structures such as /Da-/Na-/Na (1/2- 1/4-1/4 beats) and /Ta-/Ra-/Na-/Na (1/4-1/4-1/2-1/2), which were observed in our humming samples. These patterns indicate the subjects' preference for humming patterns, and bias the frequency of some specific models so that the probability of one humming note model following the other may differ. Under such circumstances, a language model could be helpful in improving the performance of the segmentation.

The figure (Figure 3) below shows a spectrogram of a typical humming input, with the output of the recognizer superposed as note labels. As can be observed in Figure 3, each note is labeled with one of the note models that are supplied in the lexicon. (SIL indicates a model for silence.)

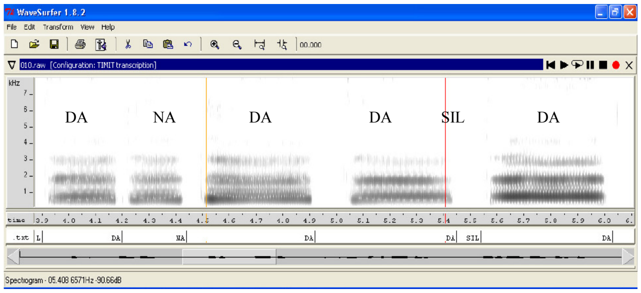
WaveSurfer 1.8.2
File Edit Transform View Help
010.raw [Configuration: TIMIT transcription]
DA
NA
DA
DA
SIL
DA
Spectrogram - 05.408 6571Hz -90.66dB

**Figure 3. The output of the recognizer**

Since the goal is to obtain accurate note segmentations, the precise identity of the hummed syllable is not critical; the only errors that affect transcription results are the note boundary errors. These errors can be considered deletions and insertions. An insertion error occurs when the recognizer defines a note boundary when, in fact, it does not exist in the input melody (see Figure 4.a). Conversely, a deletion error can be defined as a humming note that is not detected by the recognizer, and connected to the previous or next note (as shown in Figure 4.b).

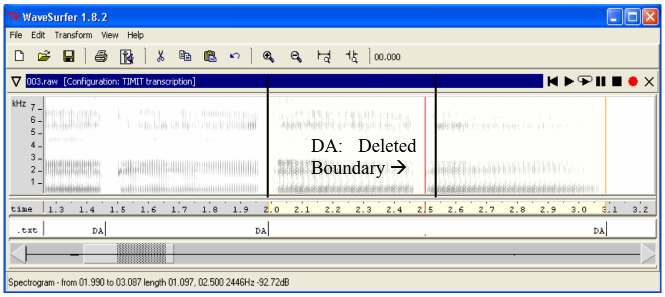
WaveSurfer 1.8.2
File Edit Transform View Help
003.raw [Configuration: TIMIT transcription]
DA: Deleted
Boundary →
DA
DA
DA
Spectrogram - from 01.990 to 03.087 length 01.097, 02.500 2446Hz -92.72dB

**Figure 4. a) Deletion: a note boundary is omitted by the system**

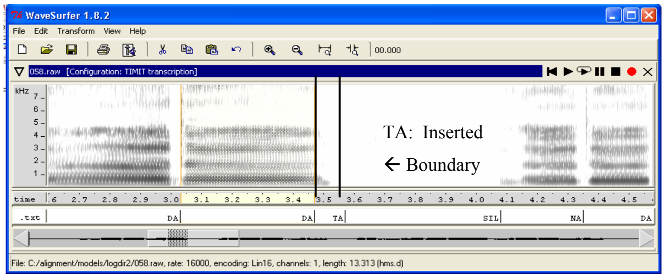
WaveSurfer 1.8.2
File Edit Transform View Help
058.raw [Configuration: TIMIT transcription]
TA: Inserted
← Boundary
DA
DA
TA
SIL
NA
DA
File: C:/alignment/models/logdir2/058.raw, rate: 16000, encoding: Lin16, channels: 1, length: 13.313 (hms.d)

**Figure 4. b) Insertion: an extra note is introduced by the system**

## 2.2 Post Processing: Energy and Pitch Analysis

Information obtained from the recognizers can contain errors, such as note insertions and deletions. To handle these errors, after the recognizer performs the initial segmentation, energy and pitch analyses are applied to the output of the recognizer. We have experimentally determined that the order of sequence of the energy and pitch analysis processing blocks does not significantly change the segmentation results.

### *2.2.1 Short-term Energy Analysis*

For each segmented humming note, the signal is windowed into frames of 20ms, with a shift of 10ms, which creates a 50% overlap between consecutive analysis frames. For each frame *k*, the short-term energy is calculated as

$$E_k = \sum_{m=1}^{N} y(m)^2 , \qquad (1)$$

where N is sampling rate × 0.02. For the energy vector *E*, an adaptive threshold value *Te* is defined by the product of the median value of *E'* (non-zero elements of the Energy vector *E*), and a constant α which was calculated from a development set. Values in *E*, greater than the threshold are quantized to 1, and the values smaller than the threshold are quantized to 0. An onset is detected if a 1-to-0 transition is followed by a 0-to-1 transition. The onset is positioned at the 0-to-1 transition point. The offset of the first note is given by the time of the 1-to-0 transition, while the offset of the second note is already defined by the recognizer.

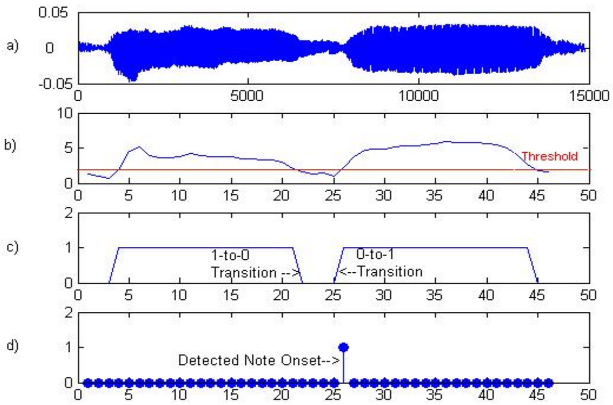

**Figure 5. a) Deletion b) Short-term energy analysis c) Energy Quantization d) Onset detection**

A deletion error in the recognizer output where two notes are concatenated and considered a single note is a common one. As shown in Figure 5, Short-term Energy Analysis is capable of detecting these errors using the aforementioned algorithm. When an onset is detected, the recognizer's output is updated with the new information.

### *2.2.2 Pitch Analysis*

Pitch tracking for the hummed signal is performed using a standard pitch detection algorithm with the PRAAT software [19].

The extracted pitch was stored in a vector *P*. For each segment, the gradient of *P* was calculated and compared to a threshold value *Tp*, which was estimated using a development set. Values that fall between the threshold regions are quantized to 0 and the rest are set to 1. An onset is determined when a 0-to-1 transition is followed by a 1-to-0 transition (see Figure 6). The onset is positioned at the time of the 1-to-0 transition.

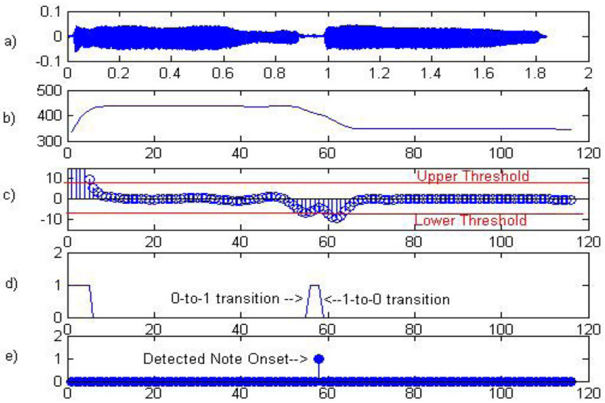

**Figure 6. a) Deletion b) Pitch contour t c) Gradient vector, and threshold region d) Pitch Quantization e) Onset detection**

After the segmentation is finalized, each note segment is assigned a single pitch value. Pitch doubling errors are corrected within each segment by searching for 1 octave differences in the max and min values of the vector *P*. In the case when pitch doubling is detected, the closest one of either max or min values to the median value is labeled as the true pitch value of that segment. If no doubling is detected, the median of *P* is assigned to that particular segment. The duration of the hummed notes are obtained directly from the extracted note boundaries as the difference between the detected offset and onset.

# 3. EXPERIMENT 1: SEGMENTATION VALIDATION

The aforementioned note segmentation method was tested with 200 actual humming samples obtained from 50 people spanning a variety of music backgrounds. The database was collected in our previous studies, as reported in [14]. Subjects were asked to use a stop-consonant syllable of their own choice for the humming. This selection was not strict that, most of the subjects preferred switching between different types of syllables during the humming of a single melody.

Two well known melodies, “Happy Birthday” and “London Bridge is Falling Down”, were selected as the target melodies for this particular study.

The actual humming database includes around 2500 humming samples of 100 people with various musical backgrounds, from those having no musical background at all to having 25+ years of professional practice. We analyzed the subject’s performance against different criteria such as the performance of humming during higher intervals vs. lower intervals, the affect of familiarity to the target melody on humming performance, etc [14].

Reference transcriptions for the hummed samples were created manually by an experienced music student. Each manual transcription was compared to the automatic transcription of the proposed front-end recognizer to evaluate the recognizer’s performance. Standard Precision (PRC) and Recall (RCL) measures are used to evaluate the segmentation performance of the system. These measures are defined as follows:

$$PRC = \frac{\textit{Number of Correctly Found Boundaries}}{\textit{Number of Hypothesized Boundaries}} \tag{2}$$

$$RCL = \frac{\textit{Number of Correctly Found Boundaries}}{\textit{Total Number of Boundaries}} \tag{3}$$

It is usually desirable to have a single evaluation value for such systems, instead of two distinct measures. In this case, the *F* measure is used. The *F* measure is a number representing the compound information of precision and recall that can be modified through appropriate weighting. The definition is of the *F* measure is as follows:

$$F = \frac{2 \times PRC \times RCL}{PRC + RCL} \tag{4}$$

A time threshold value $\Delta T$ is also proposed in order to define the location of a correct segmentation or boundary. A hypothesized boundary *t* is defined as correct if it lies within the time interval $t_0 - \Delta T < t < t_0 + \Delta T$, where $t_0$ is the correct boundary. In this study, $\Delta T$ is empirically chosen to be 75 ms (10 % of the average note length in the test dataset). *Tp* and *Te* values were estimated by using the same held out humming sample dataset.

Tests were run for 10 rounds. In each round, 10% of the data was randomly selected, and the values of *Tp* and Te (defined in section 3.2) that maximizes the *F* measure were calculated. These values of *Tp* and *Te* were then used in the tests of the remaining 90% of the data and an F measure for that particular round was calculated. After the 10 rounds were completed, results were averaged. Table 2 shows the final results.

**Table 2. Precision and recall results and relative error improvement**

| | *F* | 1 − *F* | **Error Improvement** | |
|---|---|---|---|---|
| Energy + Pitch | 0.67 | 0.33 | - | - |
| Speech Rec. | 0.79 | 0.21 | 36% | - |
| Speech Rec.+ Energy + Pitch | 0.84 | 0.16 | 51% | 23% |

*F* values for 3 experiments were calculated. The first set of segmentation experiments was performed with only energy and pitch analyses. The second set of experiments was performed with only the automatic segmentation (just the recognizer), and the third and last set of experiments was conducted using the full system, recognizer plus energy and pitch analyses. The final *F* value for the full system was found to be 0.84, which shows a

51% error improvement over Energy and Pitch, and 23% error improvement over the recognizer only performance.

# 4. EXPERIMENT 2: MELODY RETRIEVAL

Even though the proposed segmentation algorithm performs efficiently, the data itself contains uncertainty caused by variability in the humming abilities of different individuals. The details of user dependent uncertainty in humming have been discussed in detail in our earlier publication [14]. In that analysis, we showed that uncertainty in humming can be caused by reasons such as the subjects' musical background, familiarity with the melody, and ability to perceive and/or reproduce music. Because of these reasons, the audio that is produced by the subject, in the form of humming, can often carry incorrect pitch and duration information, and this incorrect reproduction may sometimes cause the melody to sound different than it should.

To further investigate the effect of user dependent uncertainty and system dependent transcription errors, retrieval experiments were performed for the same set of 200 humming samples that we used for the segmentation performance analysis. 200 humming samples were distributed equally between musically trained and non-trained subjects.

The finger printing technique based on pitch intervals and duration features that was proposed in [15] is used in this search and retrieval exercise, using a database of 1000 melodies. Fixed length pitch and duration contour information packages are extracted at certain parts of the hummed melody where the highest and lowest pitch transitions and duration ratio changes are occurred. For full description of the retrieval engine, please refer to [15].

**Table 3. Retrieval Results for the manual and automatic transcriptions**

| Retrieval | Manual Transcription | Proposed Automatic Transcription | Only Energy+ Pitch |
|---|---|---|---|
| top of the list | 83% | 76% | 63% |
| within top 5 | 88% | 83% | 71% |

Table 3 above shows the retrieval performance of our QBH system using both manually and automatically transcribed queries. Retrieval performance is measured by whether or not the system can match the input to the correct melody in the database. Usually MIR systems give a list of candidate melodies as the output. Two types of retrieval measures are proposed. The first row in the table shows the retrieval measure when the intended melody is at the top of the result list, and the second row shows the measure when the melody is found within the top N selected songs (N=5 in this case).

By inspecting Table 3, one can compare the retrieval performance of the system for manual and automatic transcriptions. With manual transcription, 83% retrieval performance is achieved. As expected 91 of the correctly retrieved queries are from musically trained subjects, and the remainder 75 are from non-trained subjects. This is consistent with our previous finding that a non-trained subject's humming contains more user dependent uncertainty than that of a trained subject. Even if one expected a musically trained subject's humming to be clear and accurate, the various sources of uncertainty may be sufficient to change the characteristics of the melody, resulting in mismatches in retrieval. This is perhaps why 100% retrieval accuracy is so illusive.

For the case where the audio to melodic symbol transcription is performed automatically, 76 percent of the input samples are retrieved correctly. Here, 83 of the correctly retrieved queries are from musically trained subjects and the rest, 69 samples, are from non-trained subjects. The performance decrease between the manual and automatic transcription can be attributed to the system's failure to accurately transcribe the audio into the desired melodic representation. Like user dependent uncertainty, system errors such as insertion and deletions during the note segmentation process can change the characteristics of the input query so that mismatches occur.

In the last column of the Table 3, retrieval results of the system using only pitch and energy analysis for the front-end is presented. Results report around 35% melody retrieval performance improvement when the front-end of the system is enhanced with the proposed HMM based transcription.

# 5. CONCLUSIONS AND FUTURE WORK

In this paper, the problem of automatic transcription of audio input to symbolic representation for use in a QBH system is discussed. A speech recognizer is used to extract the hummed notes, and its output is further refined by pitch and energy analysis. Precision, Recall and the *F* measures were calculated to measure how correctly the note boundaries are located, given a time threshold $\Delta T$ . A dataset of 200 humming files was used. With $\Delta T$ set to 75ms, an *F* measure of 0.84 was obtained. The output of the proposed system was also used in the context of retrieval performance analysis. The retrieval engine performed with 76% accuracy using the automatic segmentation output, where the accuracy is defined as finding the correct match at the top of the result table. In comparison, the manual transcriptions of the same dataset resulted in 83% retrieval accuracy.

There are several ways to further improve the system performance, one of which relates to segmentation errors. These results indicate that by improving the segmentation performance, retrieval results close to that of manual transcription can be obtained. To achieve better results, the performance of each module needs to be improved. On the speech recognizer side, better note models need to be created to decrease the frequency of deletion and insertion errors. For pitch and energy analysis, better statistical approximations of note transitions can be used.

On the other hand, the general performance of the system is also directly affected by the retrieval calculations. Better statistical models could be developed to make the retrieval algorithms more robust against user dependent uncertainty and system dependent representation errors. These are part of our ongoing work.

# 6. ACKNOWLEDGEMENTS

This work was funded in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, Cooperative Agreement No. EEC-9529152, in part by the National Science Foundation Information Technology Research Grant NSF ITR 53-4533-2720, and in part by ALi

Microelectronics Corp. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the National Science Foundation and ALi Microelectronics Corp.

## 7. REFERENCES

[1] A. Ghias, J. Logan, D. Chamberlin, and B. Smith, "Query by humming: Musical information retrieval in an audio database," in *Proc. ACM International Conference on Multimedia*, San Fransisco, California, Nov. 1995, pp. 231–236.

[2] R. J. McNab, L. A. Smith, I. H. Witten, C. L. Henderson, and S. J. Cunningham, "Towards the digital music library: Tune retrieval from acoustic input," in *Proc. ACM International Conference on Digital Libraries*, Mar. 1996, pp. 11–18.

[3] R. J. McNab, L. A. Smith, I. H. Witten, and C. L. Henderson, "Tune retrieval in multimedia library," *Multimedia Tools And Applications*, vol. 10, pp. 113–132, Apr. 2000.

[4] P. Y. Rolland, G. Raskinis, and J. G. Ganascia, "Music content-based retrieval: An overview of melodiscov approach and systems," in *Proc. ACM International Conference on Multimedia*, Orlando, Florida, Nov. 1999, pp. 81–84.

[5] S. Blackburn and D. D. Roure, "A tool for content based navigation of music," in *Proc. ACM International Conference on Multimedia*, Bristol, England, Sept. 1998, pp. 361–368.

[6] B. Chen and J.-S. R. Jang, "Query by singing," in *Proc. IPPR Conference on Computer Vision, Graphics and Image Processing*, Taiwan, Aug. 1998, pp. 529–536.

[7] N. Hu and R. B. Dannenberg, "A comparison of melodic database retrieval techniques," in *Proc. ACM International Conference on Digital Libraries*, Portland, Oregon, July 2002

[8] L. P. Clarisse, J. P. Martens, M. Lesaffre, B. D. B. amd H. De Meyer, and M. Leman, "An auditory model based transcriber of singing sequencecs," in *Proc. ISMIR International Conference on Music Information Retrieval*, France, Oct. 2002.

[9] N. Kosugi, Y. Nishihara, T. Sakata, M. Yamamuro, and K. Kushima, "Music retrieval by humming," in *Proc. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, Victoria, Canada, Aug. 1999, pp. 404–407.

[10] N. Kosugi, Y. Nishihara, T. Sakata, M. Yamamuro, and K. Kushima, "A practical query-by-humming system for a large music database," in *Proc. ACM International Conference on Multimedia*, Marina Del Rey, California, Nov. 2000.

[11] H.-H. Shih, S. Narayanan, and C.-C. J. Kuo, "An hmm-based approach to humming transcription," in *Proc. IEEE International Conference on Multimedia and Expo*, Laussanne, Switzerland, Aug. 2002, pp. 337–340.

[12] S. Pauws, "Cubyhum: A fully operational query by humming system," in *Proc. ISMIR International Conference on Music Information Retrieval*, Paris, France, October 2002.

[13] B. Pardo, Jonah Shifrin, and William Birmingham, "Name That Tune: A Pilot Study in Finding a Melody From a Sung Query," *Journal of the American Society for Information Science and Technology*, Volume 55, Issue 4, Feb. 2004, p 283-300.

[14] E. Unal, S. Narayanan, Maverick H.-H. Shih, Elaine Chew, and C.-C. Jay Kuo, "Creating data resources for designing user-centric front-ends for query-by-humming Systems," *ACM Multimedia System,* vol. 10, pp. 475-483, May 2005.

[15] E. Unal, S. Narayanan, and E. Chew, "A statistical approach to retrieval under user-dependent uncertainty in query-by-humming systems," *in Proc. ACM SIGMM MIR2004,* New York City, NY, October 2004

[16] R. Typke, F. Wiering and R. C. Veltkamp, "A Survey of Music Information Retrieval Systems," *Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR),* London, September 2005.

[17] B. Pellom, "The university of colorado continious speech recognizer." [Online]. Available: http : //cslr.colorado.edu/beginweb/speechrecognition/sonic.html

[18] "The CMU statistical language modeling (slm) toolkit." [Online]. Available: http : //www.speech.cs.cmu.edu/SLMinfo.htm

[19] Paul Boersma and David Weenink "Praat: doing phonetics by computer." [Online]. Available : http://www.fon.hum.uva.nl/praat/

[20] B. Capleton, "Perfect pitch." [Online]. Available: http ://www.amarilli.co.uk/piano/perfectp.asp

[21] R. A. Cole, J. Mariani, H. Uszkoreit, A. Zaenen and . Zue, "Survey of the state of the art in human language technology," *Center for spoken language understanding CSLU*, Carnegie Mellon University, Pittsburgh, PA.