A MUSIC RETRIEVAL SYSTEM USING CHROMA AND PITCH FEATURES BASED ON CONDITIONAL RANDOM FIELDS

Kouhei Sumi, Miki Arai, Takuya Fujishima, and Seiichi Hashimoto

Corporate Research & Development Center, Yamaha Corporation, Shizuoka, Japan

ABSTRACT

This paper presents a new symbol-based retrieval method on a polyphonic music collection which takes a sequence data of users' performances as a query. We focus on chroma and pitch features to yield a robust retrieval with queries which are generated from different arrangements and which include some mistakes. Conditional random fields (CRFs) are used to enhance simultaneous utilization of chroma and pitch features. This is because CRFs can discriminate the correct sequence from all the other candidate sequences without independence assumptions for features of the inputs. Experimental results show that the use of multiple features based on CRFs leads to a significant improvement of retrieval accuracy and accomplishes robust music retrieval regardless of performance style of queries.

***Index Terms*—** music information retrieval, machine learning, conditional random fields

1. INTRODUCTION

The purpose of this study is to develop a robust retrieval system for music signals represented in a symbolic form. In recent years, development of smart devices and cloud computing technology has allowed enormous internet users to store and share a lot of data on the servers. Also, MIDI interfaces for smart device, e.g., i-MX1[1], and MIDI Mobilizer[2], have come into the market. Accordingly, some applications have enabled music performers to upload musical audio and symbolic musical data of their recordings by connecting instruments to the Internet via smart devices, and users can freely listen to them. However, very little portion of such accumulated data will contain tags, e.g., artist name, song title, or other information. Therefore, a scheme by which users can search a specific music piece by a content-based query such as a similar phrase will be needed.

In order to realize such a content-based retrieval, we propose a new retrieval method for a kind of query-by-example (QBE) problems. Here, target queries consist of symbolic data such as MIDI, performed by users, and include both melodies and the accompaniment or either of them. Our final goal is to find the same or similar part to a given query as well as to identify the song title, artist name, and so on.

So far, the field of content-based music information retrieval (CBMIR) has developed and there have been many researches of QBE and query-by-humming (QBH). Shifrin *et al.* have proposed an HMM-based retrieval for QBH[1]. They transcribed audio data given as a query to MIDI-like representation, and extracted pitch and duration features. HMMs were trained from symbolic musical data. They reported the HMM approach obtained better results compared to the string matching approach. Although their framework treats only monophonic queries, the result indicates that the statistical modeling is effective for CBMIR.

Methods for matching polyphonic symbolic queries with a polyphonic symbolic collection have been proposed. Most of the researches focus on pitch and duration features, e.g., [2][3][4]. However, it is not always true that queries and reference data in database are played from the same music score and by the same instrumental setting. In that case, the pitch and duration features for the reference substantially differ from those for the query.

On the other hand, chroma features, namely Pitch Class Profiles (PCPs)[5], are widely used for calculating similarity in audio matching and audio-based retrieval[6][7]. They are of importance for cover song retrieval[8][9][10][11]. This is because extracted chroma features often become similar for the same songs even if the melody of the query is different from that of the reference.

In this paper, we use chroma and pitch features and statistical modeling to accomplish robust retrieval with queries performed by users. Chroma features are expected to work well when queries and reference data are played from different music score or when the queries include some mistakes. Furthermore, as statistical models we adopt conditional random fields (CRFs)[12], which are discriminative models, so as to use chroma and pitch features more effectively. Since CRFs can capture many correlated features of inputs, they allow flexible feature designs for various information. There are some researches that use CRFs on musical information processing tasks, e.g., collective annotation of music and audio-to-score matching. They reported CRFs worked better than Gaussian Mixture Models (GMMs) and Hidden Markov Models (HMMs) on these tasks[13][14].

2. STATISTICAL MODELING FRAMEWORK

In this paper, we treat the retrieval task as a labeling problem for a sequence of features obtained from queries, and propose a new framework based on chroma and pitch features extracted from symbolic data and statistical models (i.e. CRFs). Therefore, we address the problem of detecting a label sequence by using CRFs, given the input feature sequence.

2.1. Features

We use a 14-dimensional feature vector which consists of the duration ratios, each for one of 12 semitones, the top note, and the bottom note. The feature vector is calculated for each time frame of one bar period (Figure 1).

2.1.1. Pitch class duration ratio

We introduce *pitch class duration ratios* (PCDRs) which are equivalent to chroma features obtained from symbolic data. In the case

[1] http://usa.yamaha.com/products/music-production/accessories/usb-midi/i-mx1/

[2] http://line6.com/midimobilizer/

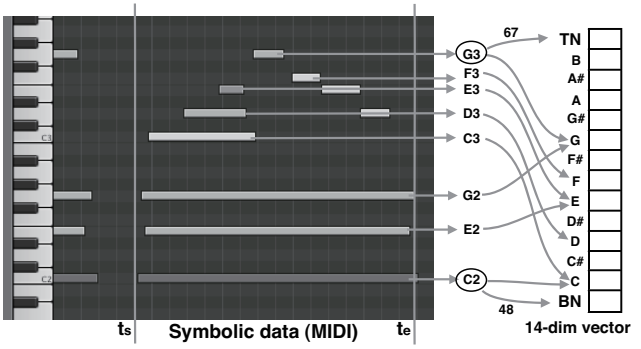

Fig. 1. Extraction of 12-semitone pitch class duration ratios, the top note, and the bottom note

of an audio signal, intensities of the semitone pitch classes can be extracted from the signal using its power spectrum, as in PCPs[5], . In the case of symbolic data, however, the intensities for the notes are not reliable. Thus, we focus on the duration of each note instead.

At a time frame t_i from t_i^s to t_i^e, a sounded note n_m^i is defined as follows:

$$n_m^i = (d_m^i, p_m^i) \quad \{m = 1, \cdots, M\} \tag{1}$$

where d_m^i is the duration time of the note (from note-on time to note-off time), p_m^i is the pitch of the note represented as the MIDI note number, and M is the number of notes. PCDR, a 12-dimentional vector, is then computed by octave invariant summation of durations of all the notes, i.e.,

$$\text{PCDR}_j^i = \frac{1}{C_i} \sum_{m=1}^{M} dur(n_m^i, j) \quad \{j = 0, \cdots, 11\} \tag{2}$$

$$dur(n_m^i, j) = \begin{cases} d_m^i & \text{if } j = p_m^i \bmod 12 \\ 0 & \text{otherwise.} \end{cases}$$

where $C_i = \sum_{m=1}^{M} d_m^i$ is a normalization factor. When the target symbolic data has a correct tempo, we calculate PCDRs for each bar. Without tempo information, PCDRs are computed for each frame of a constant period.

2.1.2. Top note and bottom note

In addition to PCDRs, we adopt the top and the bottom note information as pitch features. If some of reference data have a similar chord progression, especially in the same key, it is difficult to distinguish them when using only PCDRs. Thus, we use the top note TN_i, which is the highest note number, and the bottom note BN_i, which is the lowest note number, every frames. It can be expected that their trajectories would coincide with envelopes of the melody line and the bass line, respectively.

2.2. Conditional Random Fields

We focus here on conditional random fields (CRFs)[12], which are discriminative models applied to sequential labeling problems. CRFs are trained to discriminate the correct sequence from all the other candidate sequences without assumptions of independence for features of inputs.

2.2.1. Music labeling task

We define CRFs for music labeling as the conditional probability of an output label sequence $\mathbf{y} = (y_1, \cdots, y_n)$ given an input feature sequence $\mathbf{x} = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_\mathbf{x}} \exp\Big(\sum_{i=1}^{n} \sum_k \lambda_k f_k(y_{i-1}, y_i, \mathbf{x})\Big) \tag{3}$$

where y_i denotes one of actual labels $\{L_1, \cdots, L_N\}$ and $Z_\mathbf{x}$ is a normalization factor over all candidate paths defined as follows:

$$Z_\mathbf{x} = \sum_{\mathbf{y}' \in \{L_1, \cdots, L_N\}} \exp\Big(\sum_{i=1}^{N} \sum_k \lambda_k f_k(y'_{i-1}, y'_i, \mathbf{x})\Big) \tag{4}$$

In (3), $f_k()$ is an arbitrary feature function over i-th label y_i and its previous label y_{i-1}. $\lambda_k (\in \{\lambda_1, \cdots, \lambda_K\} \in \mathbb{R}^K)$ is a learned parameter associated with the feature function, and N is the number of labels.

For descriptive purposes, we introduce *the global feature function* $\mathbf{F}(\mathbf{y}, \mathbf{x}) = \{F_1(\mathbf{y}, \mathbf{x}), \cdots, F_K(\mathbf{y}, \mathbf{x})\}$, where $F_k(\mathbf{y}, \mathbf{x}) = \sum_{i=1}^{n} f_k(y_{i-1}, y_i, \mathbf{x})$, and we redescribe learned parameters as $\mathbf{\Lambda} = \{\lambda_1, \cdots, \lambda_K\}$. Accordingly, $P(\mathbf{y}|\mathbf{x})$ is represented as:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_\mathbf{x}} \exp(\mathbf{\Lambda} \cdot \mathbf{F}(\mathbf{y}, \mathbf{x})) \tag{5}$$

The most likely label sequence $\hat{\mathbf{y}}$ for an input sequence is obtained by maximizing this probability, i.e.,

$$\hat{\mathbf{y}} = \arg\max_\mathbf{y} P(\mathbf{y}|\mathbf{x}) = \arg\max_\mathbf{y} \mathbf{\Lambda} \cdot \mathbf{F}(\mathbf{y}, \mathbf{x}) \tag{6}$$

Since $Z_\mathbf{x}$ does not depend on label sequence, $\hat{\mathbf{y}}$ can be found with Viterbi algorithm. The graphical structure is given as an undirected graph.

2.2.2. Parameter estimation

We train CRFs by maximizing the log-likelihood $\mathcal{L}_\mathbf{\Lambda}$ of given training set $T = \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^N$,

$$\mathcal{L}_\mathbf{\Lambda} = \sum_j \log P(\mathbf{y}_j|\mathbf{x}_j) = \sum_j \big[\mathbf{\Lambda} \cdot \mathbf{F}(\mathbf{y}_j, \mathbf{x}_j) - \log(Z_{\mathbf{x}_j})\big] \tag{7}$$

When $\mathcal{L}_\mathbf{\Lambda}$ is at its maximum, the gradient of $\mathcal{L}_\mathbf{\Lambda}$ is equal to zero, i.e:

$$\nabla \mathcal{L}_\mathbf{\Lambda} = \sum_j \big[F_k(\mathbf{y}_j, \mathbf{x}_j) - E_{P(\mathbf{y}_j|\mathbf{x}_j)} F_k(\mathbf{y}, \mathbf{x}_j)\big] = 0 \tag{8}$$

where $E_{P(\mathbf{y}_j|\mathbf{x}_j)} F_k(\mathbf{y}, \mathbf{x}_j)$ is the expectation of the global feature k over the model distribution $P(\mathbf{y}|\mathbf{x})$ and T. This expectation can be computed efficiently using a variant of the foward-backward algorithm.

$$\begin{aligned} E_{P(\mathbf{y}_j|\mathbf{x}_j)} F_k(\mathbf{y}, \mathbf{x}_j) &= \sum_y P(y|\mathbf{x}) F(y, \mathbf{x}) \\ &= \sum_{y', y} \frac{\alpha_{y'} \cdot f_k^* \exp(\sum_{k'} \lambda_{k'} f_{k'}^*) \cdot \beta_y}{Z_\mathbf{x}} \end{aligned} \tag{9}$$

where f_k^* denotes $f_k(y', y, \mathbf{x})$, y and y' represent neighboring labels in the observed sequence in the training data T, and α_y and β_y are the foward and backward state-cost vectors defined as follows:

$$\alpha_y = \sum_{y' \in L(y)} \alpha_{y'} \cdot \exp\Big(\sum_k \lambda_k f_k(y', y, \mathbf{x})\Big) \tag{10}$$

$$\beta_y = \sum_{y' \in R(y)} \beta_{y'} \cdot \exp\Big(\sum_k \lambda_k f_k(y, y', \mathbf{x})\Big) \tag{11}$$

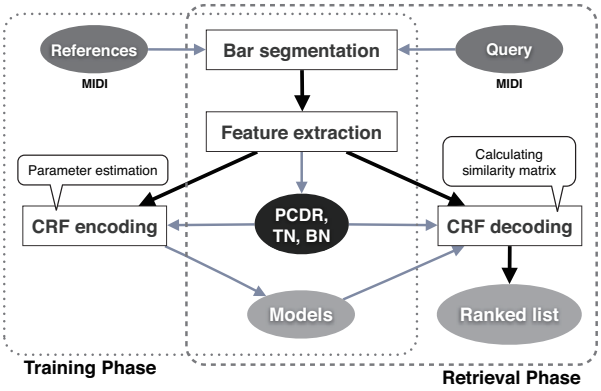

Fig. 2. System Overview

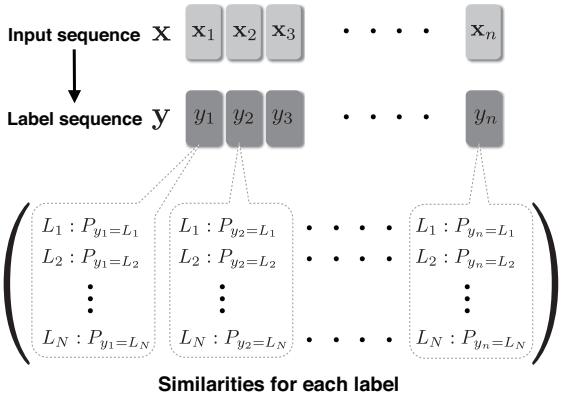

Fig. 3. Similarity matrix

where $L(y)$ and $R(y)$ denote a set of labels each of which connects to the label y from the left and the right, respectively. These costs are initialized by 1 in case of the departure and the termination state. A normalization factor is then a constant $Z_{\mathbf{x}} = \alpha_{y_{end}}$. Therefore, we can accumulate the feature expectations from calculation of these costs.

In order to avoid overfitting, it is effective to regularize the likelihood. Kudo *et al.*[15] compared two types of regularizations, the likelihood with a Gaussian prior (L2-norm) and a Laplacian prior (L1-norm). They reported L2-norm method performed slightly better than L1-norm did in a Japanese morphological analysis task. Thus, we apply regularization with L2-norm. The log-likelihood $\mathcal{L}_{\mathbf{\Lambda}}$ can be described with a Gaussian prior as follows:

$$\mathcal{L}_{\mathbf{\Lambda}} = \sum_j \log P(\mathbf{y}_j|\mathbf{x}_j) - \frac{1}{2}\sum_k |\lambda_k|^2 \tag{12}$$

and the gradient is as follows:

$$\nabla\mathcal{L}_{\mathbf{\Lambda}} = \sum_j \left[F_k(\mathbf{y}_j, \mathbf{x}_j) - E_{P(\mathbf{y}_j|\mathbf{x}_j)}F_k(\mathbf{y}, \mathbf{x}_j)\right] - \lambda_k \tag{13}$$

The optimal parameters are given when $\nabla\mathcal{L}_{\mathbf{\Lambda}}$ is equal to zero, and can be obtained by various methods such as quasi-Newton and L-BFGS methods.

3. PROPOSED MUSIC RETRIEVAL SYSTEM

An overview of our proposed retrieval system is shown in Figure 2. The following sub-sections describe the detail of each process.

3.1. Feature extraction

We use a 14-dimentional vector described above as the feature vector which is computed from symbolic musical data given as a standard MIDI file. First, we conduct bar segmentation based on tempo information. If the file does not include tempo information, we assume 120 beats per minute to detect bars. Then, PCDRs, the top note, and the bottom note are extracted for each of the detected bars.

3.2. Definition of local feature functions

We define each local feature function as follows:

- Each probability density of PCDR^i_j.
- Each joint probability density of combinations for a major triad, $(\mathrm{PCDR}^i_j, \mathrm{PCDR}^i_{j+4}, \mathrm{PCDR}^i_{j+7})$.
- Each joint probability density of combinations for a minor triad, $(\mathrm{PCDR}^i_j, \mathrm{PCDR}^i_{j+3}, \mathrm{PCDR}^i_{j+7})$.
- Unigram of the current top note TN_i.
- Unigram of the current bottom note BN_i.
- Bigram of TN_{i-1} and TN_i.
- Bigram of BN_{i-1} and BN_i.

Here, we assume that these probability densities are approximated with Gaussian distributions. They are obtained using all training sets in advance.

3.3. Ranking algorithm

In order to obtain the bar-wise retrieval result, we use marginal probabilities for all candidates as a kind of confidence measure. The marginal probability $P_{y_i=L_j}$ for i-th label y_i to be equal to L_j given an input feature sequence $\mathbf{x}$ can be computed by forward-backward algorithm using the forward and backward state-cost vectors α_y, β_y described in Section 2.2 as follows:

$$P_{y_i=L_j} = \frac{1}{Z_{\mathbf{x}}}\alpha_{y_i=L_j} \cdot \beta_{y_i=L_j} \tag{14}$$

We define a *similarity matrix* as the matrix of which element at the i-th row and j-th column is expressed as $P_{y_i=L_j}$ (Figure 3). This enables us to accomplish music retrieval for not only whole of a query but also a specific part because the ranked list for an arbitrary part can be obtained by element-wise comparison of corresponding columns in a similarity matrix.

4. EXPERIMENTAL RESULTS

We tested our system on polyphonic piano solo performances of classical and popular music. Two query sets were manually constructed by human subjects. The first set had 75 queries, played from the music scores used for the reference database (Set1). The second set had 75 queries which were in different styles from the references: some had no melody, some had no accompaniment, and some were arranged suitably (Set2). Some queries in both sets had mistakes. The average length of all queries is approximately 40 seconds, and almost all of them consist of only 16-bar or 32-bar performance. As a reference collection, we used 1422 commercially available MIDI

Table 1. Results of models and features comparison

Method		Set1		Set2		All		
Models	Features	AR	MRR	AR	MRR	AR	SD	MRR
GMMs	PCDR	11.2	0.60	21.3	0.39	16.3	36.4	0.49
GMMs	PCDR,TN,BN	18.7	0.44	22.4	0.42	20.5	40.1	0.43
HMMs	PCDR	12.6	0.56	22.6	0.37	17.6	36.0	0.46
HMMs	PCDR,TN,BN	8.6	0.56	16.6	0.46	12.6	24.3	0.51
CRFs	PCDR	7.2	0.48	9.1	0.38	8.1	9.4	0.43
CRFs	**PCDR,TN,BN**	**4.9**	**0.61**	**7.5**	**0.46**	**6.2**	**9.2**	**0.53**

files[3] which are polyphonic piano solo pieces of classical and popular music. Note that the query in test sets has the same key as the corresponding piece in the reference collection.

To evaluate the effectiveness and the robustness of the proposed method, we measure the average rank (AR), the standard deviation (SD) of ranking for all queries, and the average Mean Reciprocal Rank (MRR: 1/rank of the first correct item) of methods based on the following viewpoints of models and features.

1. Model-comparison: GMMs vs. HMMs vs. CRFs.
2. Feature-comparison: only PCDR vs. all features (PCDR, the top note, and the bottom note).

The results are listed in Table 1. The CRFs-based method with both chroma and pitch features (proposed) improved both the average rank and the average MRR in comparison to other methods. Also, a comparison of SD values indicates that the proposed method made it possible to reduce the variability of ranking. These results prove the effectiveness of using both chroma and pitch features based on CRF models. In case of using CRFs and HMMs, introduction of the top note and the bottom note improved the performance without exception. On the other hand, such feature combination in GMMs leads to a degradation of the accuracy in many cases. This is attributed to the fact that GMMs do not have tranitional information and are influenced by local difference between features.

Whereas the proposed method significantly improved the average rank compared to the HMMs-based method, the average MRR by HMMs-based method became nearly equal to that by the proposed method, especially toward Set2. This shows the results of HMMs-based vary considerably depending on the query. Thus, the result means that the proposed method achieves more robust retrieval than the other methods.

5. CONCLUSIONS

We presented a new method for symbolic music retrieval using chroma and pitch features based on CRFs. CRF models enhance simultaneous utilization of chroma and pitch features because they enable flexible feature designs for multiple information. Experimental results show that the use of both chroma and pitch features based on CRFs is more effective in symbolic music retrieval than that based on other models.

As a future work, we plan to apply our framework to audio data. Proposed PCDRs are expected to have high affinity for PCPs[5], since both are based on intensities of the semitones. Extracting pitch features corresponding to the top and the bottom note needs accurate pitch estimation from polyphonic audio. Thus, we would extend the framework in which errors in pitch estimation do not significantly affect music retrieval. We also plan to combine hashing techniques, e.g., locality sensitive hashing[11], to quickly find similar items in large databases.

[3] http://www.music-eclub.com/musicdata/

6. REFERENCES

[1] J. Shifrin, B. Pardo, C. Meek, and W. Birmingham, "HMM-based musical query retrieval," in *Proc. of Joint Conf. on Digital Libraries*, 2002, pp. 295–300.

[2] B. Pardo and M. Sanghi, "Polyphonic musical sequence alignment for database search," in *Proc. of ISMIR*, 2005, pp. 215–222.

[3] J. Pickens and C. Iliopoulos, "Markov random fields and maximum entropy modeling for music information retrieval," in *Proc. of ISMIR*, 2005, pp. 207–214.

[4] I. S. H. Suyoto, A. L. Uitdenbogerd, and F. Scholer, "Searching musical audio using symbolic queries," *J. of IEEE. Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 372–381, 2008.

[5] T. Fujishima, "Realtime chord recognition of musical sound : a system using common lisp music," in *Proc. of ICMC*, 1999, pp. 464–467.

[6] N. Hu, R. B. Dannenberg, and G. Tzanetakis, "Polyphonic audio matching and alignment for music retrieval," in *Proc. of IEEE WASPAA*, 2003, pp. 185–188.

[7] M. Müller, F. Kurth, and M. Clausen, "Audio matching via chroma-based statistical features," in *Proc. of ISMIR*, 2005, pp. 288–295.

[8] J. P. Bello, "Audio-based cover song retrieval using approximate chord sequences: Testing shifts, gaps, swaps and beats," in *Proc. of ISMIR*, 2007, pp. 239–244.

[9] D. P. W. Ellis and G. E. Poliner, "Identifying 'cover songs' with chroma features and dynamic programming beat tracking," in *Proc. of ICASSP*, april 2007, vol. 4, pp. 1429–1432.

[10] J. Serrà, E. Gómez, P. Herrera, and X. Serra, "Chroma binary similarity and local alignment applied to cover song identification," *J. of IEEE. Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 6, pp. 1138–1151, 2008.

[11] M. Casey and M. Slaney, "Fast recognition of remixed music audio," in *Proc. ICASSP*, 2007, vol. 4, pp. 1425–1428.

[12] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic modelsfor segmenting and labeling sequence data," in *Proc. of ICML*, 2001, pp. 282–289.

[13] Z. Duan, L. Liu, and C. Zhang, "Collective annotation of music from multiple semantic categories," in *Proc. of ISMIR*, 2008, pp. 237–242.

[14] C. Joder, S. Essid, and G. Richard, "A conditional random field viewpoint of symbolic audio-to-score matching," in *Proc. of ACM Multimedia*, 2010, pp. 871–874.

[15] T. Kudo, K. Yamamoto, and Y. Matsumoto, "Appliying conditional random fields to japanese morphological analysis," in *Proc. of EMNLP*, 2004, pp. 230–237.