QUERY BY SINGING/HUMMING SYSTEM BASED ON THE COMBINATION OF DTW DISTANCES FOR MIREX 2011

Dalwon Jang, Chai-Jong Song, Saim Shin, Jong-Seol Lee,
Sung-Joo Park, Sei-Jin Jang, Seok-Pil Lee, and Kyeung Hak Seo
Korea Electronics Technology Institute (KETI), Seoul, Rep. of Korea
{dalwon, jcsong, miror, leejs, bpark, sjjang, lspbio, paul_kseo}@keti.re.kr

ABSTRACT

This extended abstract descries KETI's submission to the query-by-singing/humming (QbSH) task of MIREX 2011. Our QbSH system is based on dynamic time warping (DTW) and frame-based pitch sequence. Our system reduces false alarm to combine the distances of multiple DTW processes. To improve the performance, DTW with asymmetric sense, compensation, and distances insensitive to the error are investigated.

1. INTRODUCTION

The goal of the query-by-singing/humming (QbSH) system is to retrieve songs using human's acoustic singing /humming query [1–7] . The QbSH task in MIREX 2011 has a goal to evaluate various QbSH systems. In the evaluation, two subtasks are proposed: classic QbSH evaluation and variants QbSH evaluation. The first one is based on MIDI files, and .wav format human singing/humming query. The second one constructs database based on .wav format human singing/humming snippets. In both subtasks, top-10 hit rate is only a performance measure.

2. SYSTEM DESCRIPTION

Matching engine of our QbSH system is based on the dynamic time warping (DTW) algorithm, and we incorporate the combination of DTW distance, asymmetric sense, compensation, and distances insensitive to the error. To reduce false alarm, various DTW processes are performed, and the DTW distances are combined. To match a short query with a song, asymmetric DTW is used. The method to compensate the incorrect singing/humming and the saturated distances which are not highly sensitive to the error, are also used in the system.

The input of QbSH matching engine is a set of frame-based pitch sequences. Matching engine should be robust against the mismatch between pitch sequence of query $\mathbf{S}_q$ and the pitch sequence stored in DB. From now, $\mathbf{S}_{DB}^{(i)}$ denotes the pitch sequence of ith song. The objective of matching engine can be mathematically formulated as follow:

$$\hat{i} = \arg\min_{i=1}^{I} d_M(\mathbf{S_q}, \mathbf{S}_{DB}^{(i)})$$

where $d_M(\cdot)$ is the distance computed in matching engine.

Commonly, users sing/hum at inaccurate absolute/ relative pitch with a wrong tempo [3]. To compensate pitch, brute-forth search is used. The system finds the minimum distance by changing the compensation coefficient. Mathematically,

$$\begin{aligned} d_M(\mathbf{S_q}, \mathbf{S}_{DB}^{(i)}) &= \gamma_1 \min_{c\in\mathbf{C}_1} d_{DTW}(\mathbf{S}_q + c, \mathbf{S}_{DB}^{(i)}) + \\ &\quad \gamma_2 \min_{c\in\mathbf{C}_2} d_{DTW}(\mathbf{S}_q + c, \mathbf{S}_{DB}^{(i)}) + \\ &\quad \gamma_3 d_{DTW}(\hat{\mathbf{S}}_q, \hat{\mathbf{S}}_{DB}^{(i)}) \end{aligned}$$

where c is compensation coefficient and $(\gamma_1, \gamma_2, \gamma_3)$ are weighing coefficients. The sequence $\hat{\mathbf{S}}_q$ and $\hat{\mathbf{S}}_{DB}^{(i)}$ mean delta sequence of $\mathbf{S}_q$ and $\mathbf{S}_{DB}^{(i)}$, respectively. In our system, $\mathbf{C}_1$ is set to $min(\mathbf{S}_{DB}^{(i)}) - min(\mathbf{S}_q) + \{-5, -4, \cdots, 5\}$, and $\mathbf{C}_2$ is set to $min(\mathbf{S}_{DB}^{(i)}) - min(\mathbf{S}_q) + \{-4.5, -3.5, \cdots, 4.5\}$. Thus, $d_M(\cdot)$ is the combination of three distances. Among the three, the first two are decided based on minimum search. By combining the distances, the false alarm is reduced.

Our system uses the DTW algorithm which is widely used for QbSH system since it gives the robust matching results against local timing variation and inaccurate tempo. As in [4], determining DTW path is asymmetric, and the difference is a weighting coefficient. In system proposed in [4], the weighing coefficient is 2, but we set the coefficient as 4.

The performance of QbSH system is dependent on the distance between two elements of different vectors. When using DTW, absolute difference or squared difference is commonly used. In our works, the following distance is used.

$$d_{HINGE}^{(\lambda)}(a,b) = \begin{cases} |a-b| & if |a-b| < \lambda \\ \lambda & otherwise \end{cases}$$

For the second subtask, we should keep it in mind that two pitch sequences from two singing /humming snippets may be very different for a short period. To reduce the influence of the short period error, the distance is used. In

the distance, very big difference is limited as only λ. Our system used $\lambda = 3$.

3. REFERENCES

[1] J. -S. R. Jang and M. Y. Gao, "A query-by-singing system based on dynamic programming," *International Workshop on Intelligent Systems Resolution (the 8th Bellman Continuum)*, Hsinchu, Taiwan, pp 85-89, Dec. 2000.

[2] J. -S. R. Jang and H.-R. Lee, "A general framework of progressive filtering and its application to query by singing/humming," *IEEE Trans. on Audio, Speech, and language Processing*, vol. 16, no. 2, pp 350-358, Feb., 2008

[3] Y. Zhu and D. Shasha, "Warping indexes with envelope transforms for query by humming," *In Proc. ACM SIGMOD Int. Conf. on Management of Data*, pp. 181-192, 2003

[4] H. M. Yu, W. H. Tsai, and H. M. Wang, "A query-by-singing system for retrieving karaoke music," *IEEE Trans. on multimedia*, vol. 10, no. 8, 2008, pp. 1626-1637.

[5] A. Duda, A N$\ddot{u}$rnberger, and S. Stober, "Towards query by singing/humming on audio databases,"' *Proc. ISMIR*, 2007.

[6] M. Ryyn$\ddot{a}$nen and A. Klapuri, "Query by humming of MIDI and audio using locality sensitive hashing," *ICASSP*, 2008.

[7] L. Wang, S. Huang, S. Hu, J. Liang, and B. Xu, "An effective and efficient method for query by humming system based on multi-similarity measurement fusion," *Proc. ICALIP*, 2008