1. Introduction

Most of the database systems adopted indexing methods that were based on the objective logical features of the data [1]-[3], imposed by the systems designers. To extract a particular content from the database, the users are required to have knowledge on the logical features that are used to index the data. While this characteristic is good for database systems with logical contents, it is a great disadvantage for database with subjective contents, like sounds. It will be very hard for the users, if for searching a particular content from the database the users have to specify the logical (in this case the physical) characteristics of the content. Oppositely because sounds imposed subjective psychoacoustical effects on human, it will be very helpful to build a database system, that permit the user to make queries using subjective preference oriented searching key. For example, consider a person creating sound effects for background music. The person may have an ideal sound in mind, but it is not always easy to refer the sound using its acoustical or physical features. It will be more intuitive and less stressful to mimic the sound to search for similar sounds in the database.

In this paper, we propose a sound database system, which is able to make query, using sound as the query-key. The proposed system extracts a number of parameters that characterize the timbre of a sound, which is one of factors that impose subjective impressions that differ from person to person. To deal with the different subjective preference of each user, the system is equipped with an adaptive query mechanism that gradually absorbing the preference of the user in searching the database, which will shorten the searching time.

We have proposed a content-based sound database system in the past [4]. In the previous system, the data are indexed by their spectral characteristics, which does not necessarily strongly correlate with the users’ subjective sound impressions, while in the proposed system we utilize timbre parameters that accommodate the user’s subjective preference better to employ an adaptive searching mechanism. Another significant improvements are:

1. In the previous system, the sound data are located locally, while in the proposed system the search range includes the Internet space. The seamless integration of sound data in the local storage and sound data in the Internet web sites will greatly enrich the database.
2. The automatic modification of the available sounds to generate new sounds in its idle time.

These two newly introduced functions that enable the system to deal with various demands of sounds, are a great contribution in increasing the usability of the database. “Global database” throughout this paper, refers to a database system that has the ability to actively searching the Internet for new sounds and modified them to enrich the sound collections.

2. System Outline

The outline of the system is shown in Figure 1. For the locally registered sound, the system stores the sound in the local database and the related indexes in the index database, while for the sound that was found in the Internet, the system stores the URL of the sound in the URL database. The URL search engine continuously searches the Internet for new sounds of which durations are less than 5 seconds. The seamless integration of local sound data with the ones located in the Internet and active sounds generation, will greatly enrich the database, so that it can satisfy the demands of many users with different preferences.

The system also modifies the locally existing sound by executing pitch shifting, sound filtering and sounds mixing to create new sounds using the sound modifier.

To implement the proposed system efficiently, a host-client relation is built. In this system the local sound database and the URL database is attached to a host computer. The host computer can be accessed by an arbitrary number of client computers, which keep only the parameter database locally. The task of the client system is to extract the parameters for the inputted sound and execute the index matching, while the task of the
Sound indexing

In the registration process, the system extracts 64 parameters that characterize the timbre of the inputted sound and used them for indexing the sound. The timbre parameters are extracted from the envelope of the sound that is known to generate perceptual effect for human[5]. Figure 2 shows the original sound envelope and the simplified envelope that is used for parameters extraction. The parameters are 5 envelope parameters (the lengths of attack, decay, sustain release, and the ratio of sustain level to the maximum amplitude), 15 spectral parameters, which are physical parameters of attack and sustain periods, and 44 harmonic features of the sound envelope [6].

Adaptive searching mechanism

In the searching process, the system will extract the timbre parameter from the key sound, execute the index matching and extract the sound locally or from the Internet.

The sound searching procedure is shown in Figure 3. The system executes repetitive searching mechanism, in which each query yields five sounds, which have the least distances to the key sound. The distance is defined as follows,

\[ E_i = \sum_{j=1}^{64} a_j \frac{(X_j - X_j^k)^2}{\kappa_j} \]

(1)

where \( E_i \) denotes the distance between the key sound and the candidate sound, \( a_j \) is the weight of the \( j \)-th parameter, \( X_j \) is the \( j \)-th parameter of the key sound, and \( X_j^k \) is the \( j \)-th parameter of the candidate sound. The weight \( \kappa_j \) is defined as follows:

\[ \kappa_j = \begin{cases} X_j & (X_j > 0.1) \\ 0.1 & (X_j \leq 0.1) \end{cases} \]
sound and the $k$-th sound in the database. $X_j$ and $X^k_j$ show the $j$-th parameters of the key sound and the $k$-th sound in the database, respectively. The weight of the $j$-th parameter is denoted by $a_j$.

The retrieved five sounds are presented in order with different localizations using a stereo sound diffusion system. The user selects the most preferable one from them. If the selected sound is a satisfactory one, then the searching process is terminated; otherwise, the next search will start using the selected sound as the key sound after the preference adaptation.

To adapt the subjective preference of the users, the system executes weights renewal for every searching repetition. The weight renewal is executed as follows,

$$D^j(t) = |X^k_j(t) - X_j^j(t)|$$

in which, $D^j(t)$ is the difference between the $j$-th index of the key sound and the $i$-th candidate sound at the $t$-th search. Then, the distances are ranked in ascending order. When the rank of the chosen sound is $R^j(t)$, then the weight correction is done as follows,

$$a_j(t+1) = \frac{a_j(t)}{R^j(t)}$$

$a_j(t)$ is the weight for the $j$-th index at the $t$-th search iteration. This procedure should be executed for all indexes until the distance between the chosen and key sounds becomes the smallest. Equation 3 implies that when a particular parameter of a chosen sound is ranked low, then the parameter is insignificant for the user, so the related weight should be scaled down. Oppositely when the rank of the particular parameter is high, then the related weight should be kept at a large value.

Figure 4 illustrates the execution of weight adaptation. For the purpose of simplicity, in Figure 4, it is shown that each sound is indexed by 2 timbre parameters, that are $x$ and $y$. In this example, 5 sounds are generated by the system after the key sound was given. Suppose that, the user chose Sound 2 as the most similar sound to the key sound. From Figure 3(a) it is clear that Sound 2 has the largest difference with the key sound in term of the parameter $y$, but because the user chose Sound 2, it should be closer to the key sound in the perceptual space of the user. The interpretation of the system with regard to the choice of the user is that for the given sound, the importance of the parameter $y$ is low, so that the related weight $a_y$ should be decreased. The weights renewal will be executed until the distance of the chosen sound to the key-sound (Equation 1) becomes the smallest. Changing the preference weights is equivalent to changing the scale of each parameter, so that eventually the query is executed inside the user’s preference space.

![Figure 4. Adaptive Searching Mechanism](image)

### 3. Experiments

The proposed system was implemented on Windows 2000 machine (CPU: Pentium III: 600kHz). Preliminary experiments were done to test the efficiency of the proposed system in retrieving desirable sound data, with regard to the retrieval speed and the quality of the retrieved data. Because from users’ point of views, there will no be distinction between the data that are located locally and the ones that are located in Internet, for this preliminary test, we regard that it will be sufficient to execute the test on the local database with about 1800 sounds (about 200 MB in size). The time needed for extracting the timbre parameters, and displaying 5 candidate sounds are 3 sec and 1 sec, respectively.

Experiments were done with 10 users with no specific musical expertise or technical knowledge about the proposed system. Each of the users was asked to input a sound and extract a sound that is similar to the inputted sound according to the user’s preference. Each user conducted 5 searches using 5 sounds as follows, Sound 1: whistle, Sound 2: noise (breath out to the microphone), Sound 3: human speech, Sound 4: metallic bell, Sound 5: crumpling a peace of paper.

For comparison we also conducted experiments on database system with no preference weight. In Figures 5 and 6, “adaptive” refers to the proposed system and “fixed” refers to the system with non-adaptive weights.
sound data. Unlike the conventional database systems that usually require the users to have prerequisite knowledge about the logical features of the data, our proposed system is free from such a requirement. The propose system is also equipped with an adaptive searching mechanism that automatically accommodate the user’s preference in its searching strategy, leading to shorter searching time. The automatic sound generation function and the seamless integration of the rich collection of data in the Internet to our system contributes to the limitless enrichment of the database without having to bear the cost of data storage because only textual information in the form of URL and the index of the data are required for sound data that are located in the Internet.

To realize rapid data search and efficient management, an effective data structure has to be considered. We also plan to open the system for Internet users in the near future.

5 Conclusions and Future Works

In this paper we have proposed a new sound database system that is able to accommodate the subjective preferences of the users in extracting

References