Leland Batey

This paper looks at the research of Holger Crysandt, of the Institute of Communications Engineering at Aachen University, Germany. The paper, titled "Music identification with MPEG-7", presented December 22, 2003 at the Methods and Applications for Multimedia 2004 conference of the Society of Photo-Optical Instrumentation Engineers (SPIE), covers a method of identifying an unknown segment of music. The problem being solved is the problem of how to take an unknown segment of music, of unknown recording quality, and find what is it's most likely match. The problem most often faced is that of how to derive useful information from recordings that may be of very poor quality, and how to then qualify and compare that data accurately against a database.

The method discussed in this paper hinges on using MPEG-7 as the audio representation format. This is a significant departure from the more ordinary method of using more common standards such as MPEG-3 or MPEG-4, done because the MPEG-7 standard does most of the hard work towards making data easily derivable and accessible for analysis. Additionally, Crysandt uses what he calls *AudioSignature Description Schema*, a scheme for breaking a piece of audio into discrete pieces of data that "describe" the music.

This description scheme:

> *"... consists of two matrices called 'Mean' and 'Variance'. Each matrix consists of 16 columns (default) representing 16 frequency sub-bands and approximately 1 row per second (default)."*

The exact number of rows and columns can varied to alter the resolution of the data

being gathered. From this data, Crysandt defines a *feature vector*, with as many dimensions as there is data from the description scheme. This means that for an 8 second sample, the feature vector would have 256 dimensions (2 matrices, each with 16 columns and 8 rows (1 row per second)).

The immediate problem with this is that normally, the best way to do search and comparisons on multi-dimensional data would be to use a *kd-tree* or similar (Crysandt refers to an ss-tree in the paper). However, at this point the data has **256 dimensions**, and due to the *Curse of Dimensionality*, results might be less accurate that desired. This problem is worked around using linear algebra to simplify the data down to a euclidean distance, while still preserving the it's uniqueness.

Now that the data's been reduced to a euclidean distance, it makes nearest-neighbor searches possible (and fast!). Applying a simple nearest neighbor search or range-query yields the closest matches. In tests done with prototype software and 10000 songs, the response time was less than 1 second, with an accuracy rate of 100% (even with distorted or filtered input).

While it isn't known if this *exact* technique is used by companies that provide audio-recognition services, it is undoubtedly very similar to how they achieve their results. The fact that all companies that provide this service return results so quickly, and the fact that they also provide ranked lists of the next-most likely candidates all points to a kd-tree or similar implementation. Indeed, the simplicity of this approach, and the fact that it was a usable solution 10 years ago, with hardware from 10 years ago, strengthens the case for it as a current solution.