

A GENERIC AUDIO CLASSIFICATION AND SEGMENTATION APPROACH FOR MULTIMEDIA INDEXING AND RETRIEVAL

SERKAN KIRANYAZ, AHMAD FAROOQ QURESHI AND MONCEF GABBOUJ

*Institute of Signal Processing, Tampere University of Technology,
Tampere, Finland*

E-mail: {serkan,qureshi}@cs.tut.fi, moncef.gabbouj@tut.fi

We focus the attention on the area of generic and automatic audio classification and segmentation for audio-based multimedia indexing and retrieval applications. In particular, we present a fuzzy approach towards hierarchic audio classification and global segmentation framework based on automatic audio analysis providing robust, multimodal, efficient and parameter-invariant classification over global audio segments. The input audio is split into segments, which are classified as *speech*, *music*, *fuzzy* or *silent*. The proposed method minimizes critical errors of misclassification by fuzzy region modeling, thus increasing the efficiency of both pure and *fuzzy* classification. The experimental results show that the critical errors are minimized and the proposed framework significantly improves the efficiency and the accuracy of audio-based retrievals especially in large-scale multimedia databases.

1. Introduction

Audio information often plays an essential role in understanding the semantic content of digital media. Henceforth, audio information has been recently used for content-based multimedia indexing and retrieval. Audio may also provide significant advantages over the visual counterpart especially if the content can be extracted according to human auditory perceptual system. This, on the other hand, requires efficient and generic audio (content) analysis that yields a robust and semantic classification and segmentation.

Audio content extraction via classification and segmentation enables the design of efficient indexing schemes for large-scale multimedia databases. During the recent years, there have been many studies on automatic audio classification and segmentation using several techniques and features; most of these being limited only to speech/music discrimination with a fixed segment size. As the multimedia world presents numerous content variations, there might, however, be several shortcomings of the approaches addressed so far. For instance most of the speech/music discriminators work on the digital audio signals that are in the uncompressed domain, with a fixed capturing parameter set. Obviously, large-scale multimedia databases may contain digital audio that is in different formats (compressed/uncompressed), encoding schemes (MPEG

Layer-2, MP3, AAC, ADPCM, etc.), capturing and encoding parameters (i.e. sampling frequency, bits per sample, sound volume level, bit-rate, etc.) and durations. Therefore, the underlying audio content extraction scheme should be robust (invariant) to such variations since the content is independent from the underlying parameters that the digital multimedia world presents. For example, the same content of a *speech* may be represented by an audio signal sampled at 8KHz or 32KHz, in stereo or mono, compressed by AAC or stored in (uncompressed) PCM format, lasting 15 seconds or 10 minutes, etc. Another important drawback of many existing systems is the lack of variable-size global segmentation. An efficient and more natural solution is to extract global segments within which the content is kept stationary so that the classification method can achieve an optimum performance within the segment.

Although audio classification has been mostly realized in the uncompressed domain, with the emerging MPEG audio content, several methods have been reported for audio classification on MPEG-1 (Layer 2) encoded audio bit-stream [5]. The last years have shown a widespread usage of MPEG Layer 3 (MP3) audio [1], [5], as well as proliferation of several video content carrying MP3 audio. The ongoing research on perceptual audio coding yields a more efficient successor called (MPEG-2/4) Advanced Audio Coding (AAC) [1]. AAC has various similarities with its predecessor but promises significant improvement in coding efficiency. In previous works [6], [8], we introduced an automatic segmentation and classification method over MP3 (MPEG-1, 2, 2.5 Layer-3) and AAC bit-streams.

Most of the existing systems do not have a multimodal structure. That is, they are either designed in *bit-stream* mode where the bit-stream information is directly used (without decoding) for classification and segmentation, or in *generic* mode where the temporal and spectral information is extracted from the PCM samples and the analysis is performed afterwards. Usually, the former case is applied for improved computational speed and the latter for higher accuracy. A multimodal structure, which supports both modes (possibly to some extent) is obviously needed in order to provide feasible solutions for the audio-based indexing of large-scale multimedia databases.

In order to overcome the aforementioned problems and shortcomings, in this paper we propose a generic audio classification and segmentation framework especially suitable for audio-based multimedia indexing and retrieval systems. The proposed method has a multimodal structure, which supports both *bit-stream* mode for MP3 and AAC audio, and *generic* mode for any audio type and format. In both modes, once a common spectral template is formed from the input audio source, the same analytical procedure is performed afterwards. The spectral template is obtained from MDCT coefficients of MP3 granules or AAC frames in *bit-stream* mode and hence called as MDCT template. The power spectrum obtained from FFT of the PCM samples within temporal frames forms the spectral template for the *generic* mode.

The proposed approach has been integrated into the MUVIS system [2], [4], [7]. The proposed method is automatic and uses no information from the video signal. It also provides robust (invariant) solution for the digital audio files with various capturing/encoding parameters and modes such as sampling frequencies (i.e. 8KHz up to 48 KHz), channel modes (i.e. mono, stereo, etc.), compression bit-rates (i.e. 8kbps up to 448kbps), sound volume level, file duration, etc. In order to increase accuracy, a *fuzzy* approach has been integrated within the framework.

In order to improve the performance and most important of all, the overall accuracy, the classification scheme produces only 4 class types per audio segment: *speech*, *music*, *fuzzy* or *silent*. *Speech*, *music* and *silent* are the *pure* class types. The class type of a segment is defined as *fuzzy* if either it is not classifiable as a pure class due to some potential uncertainties or anomalies in the audio source or it exhibits features from more than one pure class. In MUVIS system [2], [4], [7], for audio based indexing and retrieval, a pure class content is only searched throughout the associated segments of the audio items in the database having the same (matching) pure class type, such as *speech* or *music*. All *silent* segments and *silent* frames within *non-silent* segments are discarded from the audio indexing. As mentioned earlier, special care is taken for the *fuzzy* content, that is, during the retrieval phase, the *fuzzy* content is compared with all relevant content types of the database (i.e. *speech*, *music* and *fuzzy*) since it might, by definition, contain a mixture of pure class types, background noise, aural effects, etc. Therefore, for the proposed method, any erroneous classification on pure classes is intended to be detected as *fuzzy*, so as to avoid significant retrieval errors (mismatches) due to such potential misclassification. In this context, three prioritized error types of classification, illustrated in Figure 1 are defined:

- **Critical Errors:** These errors occur when one pure class is misclassified into another pure class. Such errors significantly degrade the overall performance of an indexing and retrieval scheme.
- **Semi-critical Errors:** These errors occur when a *fuzzy* class is misclassified as one of the pure class types. These errors moderately affect the performance of retrieval.
- **Non-critical Errors:** These errors occur when a pure class is misclassified as a *fuzzy* class. The effect of such errors on the overall indexing/retrieval scheme is negligible.

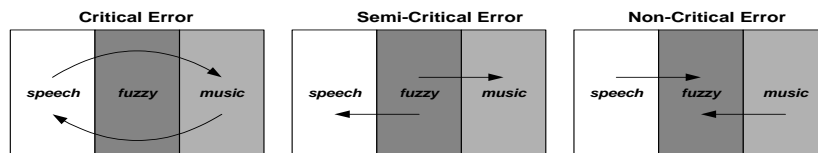


Figure 1. Different error types in classification.

2. Spectral Template Formation

In this section, we focus on the formation of the generic spectral template, which is the initial and pre-requisite step in order to provide a multimodal solution. As shown in Figure 2, the spectral template is formed either from the MP3/AAC encoded bit-stream in *bit-stream* mode or the power spectrum of the PCM samples in the *generic* mode. Basically, this template provides spectral domain coefficients, $SPEQ(w, f)$, ($MDCT$ coefficients in *bit-stream* mode or power spectrum in *generic* mode) with the corresponding frequency values $FL(f)$ for each granule/frame. Once the common spectral template is formed the granule features can be extracted accordingly and thus, the primary framework can be built on a common basis, independent from the underlying audio format and the mode used.

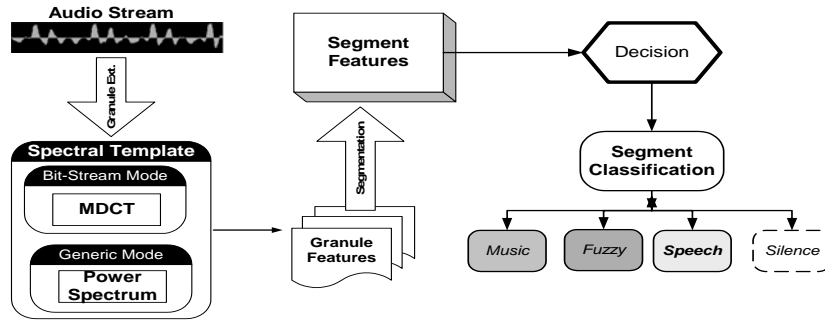


Figure 2. Classification and Segmentation Framework.

2.1. Formation of MDCT Template from MP3/AAC Bit-Stream

The *bit-stream* mode uses the compressed domain audio features in order to perform classification and segmentation directly from the compressed bit-stream. Audio features are extracted using the common *MDCT* sub-band template. The details on the formation of *MDCT* template from MP3/AAC bit-stream can be found in [8].

2.2. Formation of Spectral Template in Generic Mode

In the *generic* mode, the spectral template is formed from the FFT of the PCM samples within a frame that has a fixed temporal duration. In *bit-stream* mode, the frame (temporal) duration varies since the granule/frame size is fixed (i.e. 576 in MP3, 1024 in AAC long window mode). However, in this mode, we have the possibility to extract both fixed-size or fixed-duration frames depending on the feature type. For analysis compatibility purposes, it is a common practice to fix the (analysis) frame duration. If, however, fixed spectral

resolution is required (i.e. for fundamental frequency estimation), the frame size (hence the FFT window size) can also be kept constant by increasing the frame size by zero padding or simply using the samples from neighbor frames.

As shown in Figure 3, the *generic* mode spectral template consists of a variable size power spectrum double array $PSPQ(w, f)$ along with a variable size frequency line array $FL(f)$, which represents the real frequency value of each row entry in the $PSPQ$ array. The index w represents the window number and the index f represents the line frequency index. In generic mode, $NoW = 1$ and NoF is the number of frequency lines within the spectral bandwidth: $NoF = 2^{(\text{int}) (\log_2 (fr_{dur} f_s) - 1)}$ where fr_{dur} is the duration of one audio (analysis) frame and f_s is the sampling frequency.

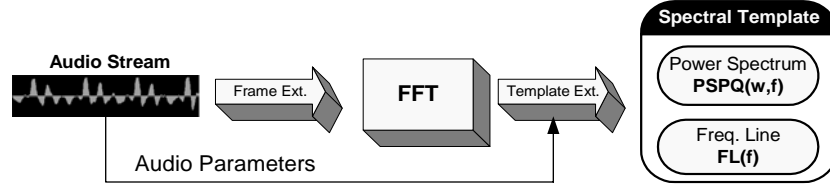


Figure 3. *Generic* Mode Spectral Template Formation.

Note that for both modes, the template is formed independently from the number of channels (i.e. stereo/mono) in the audio signal. If the audio is stereo, both channels are averaged and used as the signal before the frame extraction is processed.

3. Feature Extraction

As shown in Figure 2, a hierarchic approach has been adopted for the overall feature extraction scheme in the proposed framework. First the frame (or granule) features are extracted using the spectral template and the segment features are derived afterwards in order to accomplish classification for the segments. In the following sub-sections we will focus on the extraction of the several frame features.

3.1. Frame Features

Granule features are extracted from the spectral template, $SPEQ(w, f)$ and $FL(f)$, where $SPEQ$ can be assigned to $MDCT$ or $PSPQ$ depending on the current working mode.

3.1.1. TFE Calculation

TFE can be calculated using Eq. (1). It is the primary feature to detect *silent* granules/frames. Silence detection is also used for the extraction of TR, which is one of the main segment features.

$$TFE_j = \sqrt{\sum_w^{NoW} \sum_f^{NoF} (SPEQ_j(w, f))^2} \quad (1)$$

3.1.2. BER Calculation

BER is the ratio between the total energies of two spectral regions that are separated by a single cut-off frequency. The spectral regions fully cover the spectrum of the input audio signal. Given a cut-off frequency value f_c ($f_c \leq f_{BW}$), let $f\langle f_c \rangle$ be the line frequency index where $FL(f\langle f_c \rangle) \leq f_c < FL(f\langle f_c \rangle + 1)$, BER for a granule/frame j can be calculated using Eq. (2).

$$BER_j(f_c) = \frac{\sqrt{\sum_w^{NoW} \sum_{f=0}^{f\langle f_c \rangle} (SPEQ_j(w, f))^2}}{\sqrt{\sum_w^{NoW} \sum_{f=f\langle f_c \rangle}^{NoF} (SPEQ_j(w, f))^2}} \quad (2)$$

3.1.3. FF Estimation

If the input audio signal is harmonic over a fundamental frequency (i.e. there exists a series of major frequency components that are integer multiples of a fundamental frequency), the real FF value can be estimated from the spectral coefficients ($SPEQ(w, f)$). Therefore, we apply an adaptive peak-detection algorithm over the spectral template to check whether sufficient number of peaks around the integer multiple of a certain frequency (a candidate FF value) can be found or not. Once the peak detection is completed, a reasonable number (i.e. 5) of candidate FF peaks are obtained via HPS [3]. These candidates are then verified for harmonicity to extract the real FF value of the granule/frame.

Figure 4 illustrates a sample peak detection applied on the spectral coefficients of an audio frame with the sampling frequency 44100 Hz. Therefore, $f_{BW} = 22050 \text{ Hz}$ but the sketch shows up to around 17000 Hz for the sake of illustration. The lower subplot shows the overall peaks detected in the first step, 5 candidate peaks extracted in the second step via HPS algorithm, the multiple peaks found and finally the FF value estimated accordingly ($FF = FL(18) = 1798 \text{ Hz}$ in this example).

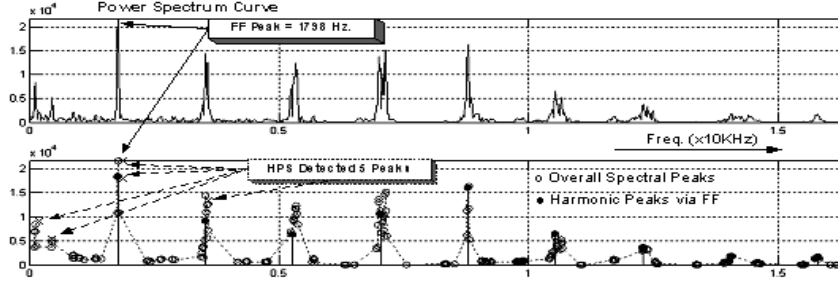


Figure 4. FF detection within a harmonic frame.

3.1.4. *SC Frequency Estimation*

SC is the first moment of the spectral distribution (spectrum) or in compressed domain it can be estimated as the balancing frequency value for the absolute spectral values. Using the spectral template arrays, *SC* frequency (f_{SC}) can be calculated using Eq. (3).

$$f_{SC} = \frac{\sum_w^{NoW} \sum_f^{NoF} (SPEQ(w, f) \times FL(f))}{\sum_w^{NoW} \sum_f^{NoF} SPEQ(w, f)} \quad (3)$$

3.2. *Segment Features*

Segment Features are extracted from the frame (or granule) features and mainly used for the classification of the segment. A segment, by definition, is a temporal window, which lasts a certain duration within an audio clip. There are basically two types: *silent* and *non-silent* segments. The *non-silent* segments are subject to further classification using their segment features; which are presented in the following sections.

3.2.1. *Dominant BER (DBER)*

For each *non-silent* segment, the dominant classifier type; the greater number of granule/frame types based on *BER* (frame feature) classification, will determine the segment type.

3.2.2. *Transition Rate (TR) vs. Pause Rate (PR)*

PR is a well-known feature as a speech/music discriminator and basically it is the ratio between the numbers of *silent* granules/frames to total number of granules/frames in a *non-silent* segment. Due to the natural pauses present in

speech content, *PR* usually achieves a significant performance in discriminating *speech* from *music*. However, as *PR* is only related with the amount (number) of *silence* (*silent* frames) within a segment, its performance is degraded when there is a fast *speech* (without sufficient amount of pauses), a background noise or when the *speech* segment is quite short (i.e. < 3s).

These erroneous cases lead us to introduce an improved measure, *TR*, which is based on the transitions, occurs between consecutive frames. *TR* can be formulated for a segment as in Eq. (4), where *NoF* is the number of frames within segment *S*, *i* is the frame index and TP^i is the transition penalization factor that can be obtained from the Table 1. Note that although the total amount of *silent* frames is low for a fast *speech* or in short *speech* segment, the transition rate will be still high due to their frequent occurrence.

$$TR(S) = \frac{NoF + \sum_i^{NoF} TP^i}{2 NoF} \quad (4)$$

Table 1: Transition Penalization Table.

Transition: $fr^i \longrightarrow fr^{i+1}$	TP^i
<i>silent</i> \rightarrow <i>non-silent</i>	+1
<i>non-silent</i> \rightarrow <i>silent</i>	+1
<i>silent</i> \rightarrow <i>silent</i>	+1
<i>non-silent</i> \rightarrow <i>non-silent</i>	-1

3.2.3. *FF* Segment Feature

FF is another well-known music/speech discriminator due to the fact that *music* is more harmonic than the *speech* in general as unlike *music* content, *speech* inherently consists of inharmonic consonants, the natural pauses and the low-bounded *FF* values (i.e. <500 Hz). Therefore, the average *FF* value within a *speech* segment tends to be quite low and vice versa for *music*.

In order to improve the discrimination factor from *FF* segment feature, we develop an enhanced segment feature based on conditional mean, which basically verifies strict *FF* tracking (continuity) within a window. Therefore, *FF* value of a particular frame will be introduced in the mean summation only if its nearest neighbors are also harmonic, otherwise discarded. The conditional mean based *FF* segment feature is formulated in Eq. (5), where FF_i is the *FF* value of the i^{th} frame in segment *S* and *j* represents the index of the frames in the nearest neighbor frame set of the i^{th} frame $NN(i)$.

$$FF(S) = \frac{\sum_i^{NoF} \left(\begin{array}{l} FF_i \text{ if } FF_j \neq 0 \forall j \in NN(i) \\ 0 \text{ otherwise} \end{array} \right)}{NoF} \quad (5)$$

Due to frequent discontinuities in the harmonicity such as pauses (*silent* frames) and consonants on a typical *speech* segment, the conditional mean results in a significantly low *FF* segment value for pure *speech* and vice versa for *music*.

3.2.4. *SC Segment Feature*

Due to the fact that unlike music, *speech* segments contains both voiced (vowels) and unvoiced (consonants), the *SC* segment feature used to perform classification is the standard deviation alone with one exception: The mean of *SC* within a segment is only used when it gives such a high value (forced-classification by *SC*) indicating the presence of the music with a certainty.

Both of *SC* segment features are extracted by smoothly sliding a short window through the frames of the non-*silent* segment. The standard deviation of the *SC* as given in Eq. (6), is calculated using local windowed mean and windowed standard deviation of *SC* in the segment, where μ_i^{SC} is the windowed *SC* mean of the i^{th} frame calculated within a window W_i with NoW frames. $\sigma^{SC}(S)$ is the *SC* segment feature of the segment S with NoF frames.

$$\sigma^{SC}(S) = \sqrt{\frac{\sum_j^{NoF} (SC_j - \mu_j^{SC})^2}{NoF}} \quad \text{where } \mu_i^{SC} = \frac{\sum_{j \in W_i}^{NoW} SC_j}{NoW} \quad (6)$$

3.3. *Perceptual Modeling in Feature Domain*

The primary approach in the classification and segmentation framework is based on the perceptual modeling in the feature domain that is mainly applied on to the major segment features: *FF*, *SC* and *TR*. Depending on the nature of the segment feature, the model provides a perceptual-rule based division in the feature space.

The forced-classification occurs if that particular feature results such an extreme value that perceptual certainty about content identification is occurred. Therefore, it overrides all the other features so that the final decision is made with respect to that feature alone. Note that the occurrence of a forced classification, its region boundaries and its class category depends on the nature of the underlying segment feature. *TR* has a forced *speech* classification region above 15%, as only pure *speech* can yield such a high value within a segment.

Similarly, *FF* has a forced *music* classification with respect to its mean value that is above 2 KHz due to the fact that only pure and excessively harmonic *music* content can yield such an extreme mean value. *SC* has two forced-classification regions, one for *music* and the other for *speech* content. The forced *music* classification occurs when the *SC* mean exceeds 2 KHz and the forced *speech* classification occurs when the primary segment feature of *SC*, the adaptive σ^{SC} value, exceeds 1200 Hz.

The region below forced-classification is where the natural discrimination occurs into one of the pure classes such as *speech* or *music*. For all segment features the lower boundary of this region is tuned so that the feature would have a typical value that can be expected from a pure class type but still quite far away having a certainty to decide the final classification alone. Finally there may be a *fuzzy* region where the feature value is no longer reliable due to various possibilities such as the audio class type is not pure, rather mixed or some background noise is present causing ‘blurring’ on the segment features. So for those segment features that are examined and approved for the *fuzzy* approach (*FF* and *SC*), a *fuzzy* region is formed and tuned experimentally to deal with such cases. Although *TR* can achieve probably the highest reliability of distinguishing *speech* from the other class types, it is practically blind of categorization of any other *non-speech* content (i.e. *fuzzy* from *music*, *music* from *speech* with significant background noise, etc.). Therefore, *fuzzy* modeling is not applied to *TR* segment feature to prevent such erroneous cases.

4. Generic Audio Classification and Segmentation

The proposed approach is mainly developed based on the aforementioned fact: automatic audio segmentation and classification are mutually dependent problems. A good segmentation requires good classification and vice versa. Therefore, without any prior knowledge or supervising mechanism, the proposed algorithm proceeds in an iterative way, starting from granule/frame based classification and initial segmentation, the iterative steps are carried out until a global segmentation and thus a successful classification per segment can be achieved at the end. Figure 5 illustrates the 4-steps iterative approach to the audio classification and segmentation problem.

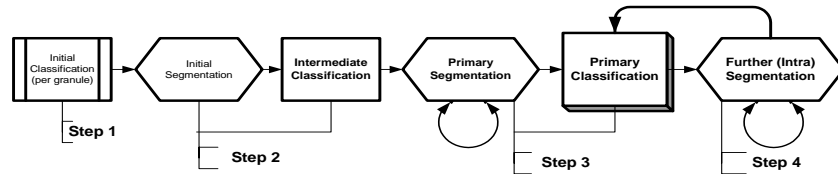


Figure 5. The flowchart of the proposed approach.

4.1. Step 1: Initial Classification

In this step, each granule/frame is classified in one of three categories: *speech*, *music* or *silent*. Silence detection is performed per granule/frame by applying a threshold (T_{TFE}) to the total energy as given in Eq. (1). T_{TFE} is calculated adaptively in order to take the audio sound volume effect into account. Further details of this process can be found in [8].

If a granule/frame is not classified as *silent*, the *BER* is then calculated for a cut-off frequency of 500 Hz due to the fact that most of *speech* energy is concentrated below 500Hz. If *BER* value for a frame is over a threshold (i.e. 2%) that granule/frame is classified as *music*, otherwise *speech*. Figure 6 illustrates step 1.

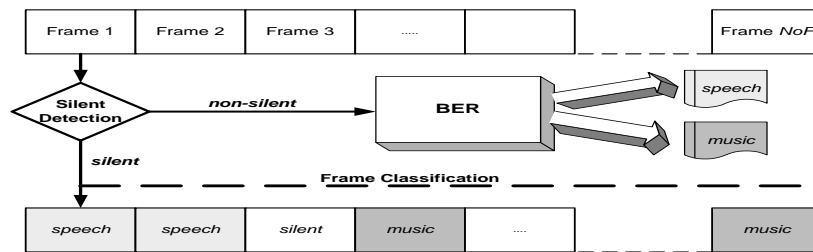


Figure 6. Step 1.

4.2. Step 2

In this step, *silent* and *non-silent* segmentations are performed on the all the *silent* granules/frames have already been found in the previous step. So the *silent* granules/frames are merged to form *silent* segments. An empirical minimum interval (i.e. 0.2 sec.) is used to assign a segment as a *silent* segment if sufficient number of *silent* granules/frames merges to a segment, which has the duration greater than this threshold. All parts left between *silent* segments can then be considered as *non-silent* segments. Once all *non-silent* segments are formed, then the classification of these segments is performed using *DBER* and *TR*. Figure 7 illustrates the process in step 2.

4.3. Step 3

The initial part of this step is dedicated to merging the small and negligible (ordinary local pauses during a natural *speech* or the borderline from one segment to another with a different class type) *silent* segments extracted in the previous step. To yield a better (global) segmentation and thus classification, all such segments (less than a threshold) are merged into the matching (according to *DBER* and *TR*). This process produces new merge-able segments, thus an iteration loop is needed until all small *silent* segments are eliminated and

non-silent segments are merged to have global segments, which have a unique classification type. The operations performed in step 3 are illustrated in Figure 8. Further details on steps 2 and 3 are available in [6], with the exception that *TR* is used instead of *PR*.

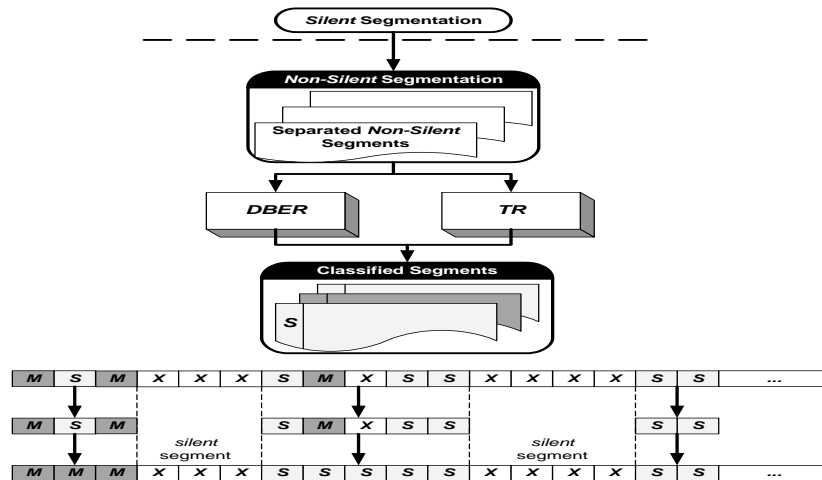


Figure 7. Step 2

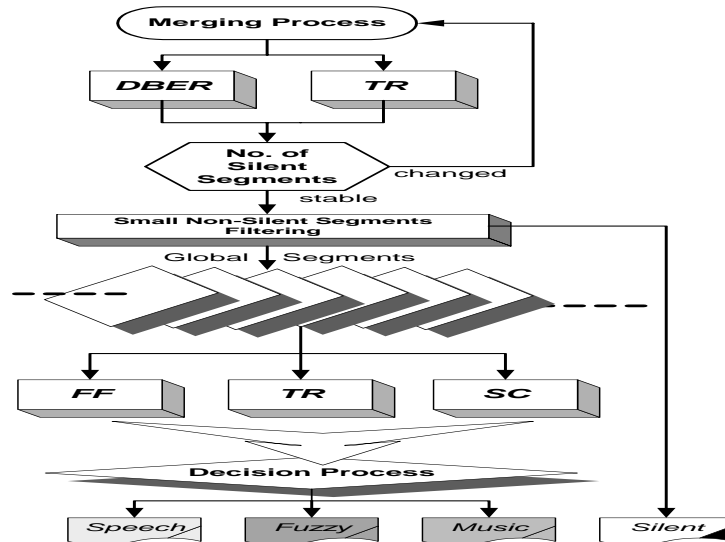


Figure 8. Step 3

Due to the perceptual modeling in the feature domain any segment feature may fall into forced-classification region and overrides the common decision process with its decision alone. Otherwise considering all possible class type combinations, a majority (result of segment features) based decision look-up table is applied for the final classification. If a common consensus cannot be made, then the segment is set as *fuzzy*.

4.4. Step 4

This step is dedicated to the intra segmentation analysis. Once final classification and segmentation is finished in step 3, *non-silent* segments with significantly long duration might still need to be partitioned into new segments if they consist of two or more sub-segments (without any *silent* part in between) with different class types. For example within a long segment there might be sub-segments that include both pure *music* and pure *speech* content without a *silent* separation in between. A series of methods are performed to successfully detect the breakpoint within such sufficiently long *non-silent* segments. This breakpoint is the real limit between two different sub-segments without a *silent* part between them. Further details on inner-breakpoint detection algorithm are available in [6].

5. Experimental Results

The evaluation of the proposed framework is carried out on the standalone MP3, AAC audio clips, AVI and MP4 files containing MPEG-4 video along with MP3, AAC, ADPCM (G721 and G723 in 3-5 bits/sample) and PCM (8 and 16 bits/sample) audio. These files contain diverse contents from TV channels showing News, Cartoon, Talk Shows, Music Clips and Commercials, or ordinary MP3 clips downloaded from Internet. Experimental music clips contain Classical, Techno, Rock, Metal, Pop and some other music types. The *speech* language is distributed among English, Turkish, French, Swedish, Arabic and Finnish languages. The duration of clips are varying between 1-5 minutes up to 2 hours. The clips are captured using several sampling frequencies from 16 KHz to 44.1 KHz so that both MPEG 1 and MPEG 2 phases are tested for Layer 3 (MP3) audio. Both MPEG-4 and MPEG-2 AAC are recorded with the *Main* and *Low Complexity* profiles (object types). TNS (Temporal Noise Shaping) and M/S coding schemes are disabled for AAC. Around 70% of the clips are stereo and the rest is mono. Total number of files used in the experiments is above 500 and in total measures, the method is applied onto 260 (> 15 hours) MP3, 100 (> 5 hours) AAC and 200 (> 10 hours) PCM (uncompressed) audio clips. The algorithm and testing conditions have been kept same in order to test the robustness of the algorithm. The error distributions results, which belong to both

bit-stream and *generic* modes, are provided in Table 2 and Table 3.

Table 2. Error Distribution for *Bit-Stream* Mode.

	<i>Speech</i>		<i>Music</i>		<i>Fuzzy</i>
	Critical	Non-Critical	Critical	Non-Critical	Semi-Critical
<i>MP3</i>	2.0 %	0.5 %	5.8 %	10.3 %	24.5 %
<i>AAC</i>	1.2 %	0.2 %	0.5 %	8.0 %	17.6 %

Table 3. Error Distribution for *Generic* Mode.

	<i>Speech</i>		<i>Music</i>		<i>Fuzzy</i>
	Critical	Non-Critical	Critical	Non-Critical	Semi-Critical
	0.7 %	4.9 %	5.1 %	22.0 %	23.4 %

From the error distributions shown in Table 2 and Table 3, it can be seen that the primary objective of the proposed scheme i.e. minimizing the critical errors, is successfully achieved. The semi-critical errors in spite of having relatively higher values, are still useful, especially considering the fact that the contribution of *fuzzy* content towards the overall size of a multimedia databases (also in experimental database) is normally less than 2%. The moderately valued non-critical errors, as the name suggests, are not critical with respect to the audio-based multimedia retrieval performance because of the indexing and retrieval scheme as discussed in detail in 2.

6. Conclusions and Future Work

In this paper we have presented a study on automatic audio content analysis and a generic framework developed for the audio-based indexing and retrieval. We have focused our efforts on providing the structural details of a generic, robust, unsupervised and multimodal system so as to accomplish a perceptual rule-based approach. We have achieved good results with respect to our primary goal of being able to minimize the critical errors on audio content classification by introducing *fuzzy* modeling on the feature domain and shown the important role of having the global and perceptually meaningful segmentation on the accurate classification (and vice versa) in this context.

The proposed work achieves significant advantages and superior performance over existing approaches for automatic audio content analysis especially in the context of audio-based indexing and retrieval for large-scale multimedia databases. Future work will focus on improving the intra segmentation algorithm and further reduction of semi-critical and non-critical error levels.

7. References

1. Karl-Heinz Brandenburg, "MP3 and AAC Explained", *AES 17th International Conference*, Florence, Italy, September 1999.
2. Moncef Gabbouj, Serkan Kiranyaz, Kerem Caglar, Esin Guldogan, Olcay Guldogan and Farooq Ahmad Qureshi, "Audio-based Multimedia indexing and retrieval scheme in MUVIS framework", *International Symposium On Intelligent Signal Processing and Communication Systems (ISPACS)*, Awaji Island, Japan, 2003.
3. M. Noll, "Pitch Determination of Human Speech by the Harmonic Product Spectrum, the Harmonic Sum Spectrum, and a Maximum Likelihood Estimate", *In Proc. Of the Symposium on Computer Processing Communications*, pp. 770-797, Polytechnic Inst. Of Brooklyn, 1969.
4. MUVIS webpage. <http://muvis.cs.tut.fi/>
5. Pan, "A tutorial on MPEG/Audio Compression", *IEEE Multimedia*, pp 60-74, 1995.
6. S. Kiranyaz, A.F. Qureshi, M.Gabbouj, "A fuzzy approach towards perceptual classification and segmentation of MP3/AAC audio", *International Symposium on Control, Communications and Signal Processing*, pp. 727-730, Hammamet, Tunisia, March 2004.
7. S. Kiranyaz, K. Caglar, O. Guldogan, and E. Karaoglu, "MUVIS: A Multimedia Browsing, Indexing and Retrieval Framework", *Proc. Third International Workshop on Content Based Multimedia Indexing, CBMI 2003*, Rennes, France, 22-24 September 2003.
8. S. Kiranyaz, M. Aubazac, M. Gabbouj, "Unsupervised Segmentation and Classification over MP3 and AAC Audio Bit-streams", *WIAMIS Workshop*, pp. 338-345, London, 2003.