석사학위논문

Master’s Thesis

상위 레벨 음악 특성을 사용한 음악 감정 분류 성능 향상

Performance Improvement of Music Mood Classification Using Hyper Music Features

최가현 (崔嘉晛 Choi, Kahyun)

정보통신공학과 디지털미디어전공

Department of Information and Communications Engineering

Digital Media Program

KAIST

2010

KOREA ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY
KAIST
한국과학기술원
SINCE 1971

상위 레벨 음악 특성을 사용한 음악 감정 분류 성능 향상

Performance Improvement of Music Mood Classification Using Hyper Music Features

Performance Improvement of Music Mood Classification Using Hyper Music Features

Advisor : Professor Minsoo Hahn

by

Kahyun Choi

Department of Information and Communications Engineering

Digital Media Program

KAIST

A thesis submitted to the faculty of the KAIST in partial fulfillment of the requirements for the degree of Master of Science in Engineering in the Department of Information and Communications Engineering, Digital Media Program.

KAIST

Daejeon, Korea

2009. 12. 18.

Approved by

Prof. Minsoo Hahn

Major Advisor

상위 레벨 음악 특성을 사용한 음악 감정 분류 성능 향상

최가현

위 논문은 한국과학기술원 석사학위논문으로

학위논문심사위원회에서 심사 통과하였음.

2009 년 12 월 18 일

심사위원장 한민수 (인)

KAIST

심사위원 최명선 (인)

심사위원 정상배 (인)

MICE
20064611

최 가 현. Choi, Kahyun.
Performance Improvement of Music Mood Classification Using Hyper Music Features. 상위 레벨 음악 특성을 사용한 음악 감정 분류 성능 향상. Digital Media Program, Department of Information and Communications Engineering. 2009. p. 55. Advisor: Prof. Hahn, Minsoo. Text in English.

Abstract

When people want to find music, they traditionally search it with its related symbolic information, such as title, lyrics, and name of the artist. As the digital music database becomes massive, however, it is not effective to rely only on those conventional queries for finding a specific song from the huge music database, because the user often forget the title or name of the artist. Moreover, it is getting common that the users want to be recommended a contextually proper playlist. Therefore, many polished music information retrieval techniques have developed so far, for instance, query by humming or tapping, finding similar songs to the seed songs, recommend songs with specific mood and genre. It is clear that those automated music search systems are heavily based on automatic music classification. It is almost impossible to manually extract important features and classify them with a database of thousands of songs, which is relatively small size though. This thesis deeply concerns audio music mood classification (AMC) which plays a key role in one of the most promising next generation music exploring systems.

In order to take mood into account for the AMC, we should formulate the vague concept, mood. After that, it is required that reliable mappings between songs and moods based on human assessment. To fulfill the requirement for trustworthy research results, we adapt five mood classes, which were defined and verified in MIREX (Music Information Retrieval Evaluation eXchange). Similarly, we also used

KAIST

600 mood-labeled music data which MIREX offers and uses for the contest.

For the similar reasons, we used MARSYAS for the reference system. MARSYAS, the most famous music information retrieval system, contains well-known music features and Support Vector Machine (SVM) classifier. It is a universal system, but it ranked the first and second in the MIREX AMC tasks, respectively.

In this thesis, mid-level music features are introduced. To explore the necessity of feature extraction process we carefully optimized SVM with barely processed signal, and then compare the results with the introduced features. Then, we expanded the relatively low-level feature set, which is used in MARSYAS, by appending the proposed mid-level features.

The newly proposed mid-level features in this thesis are chord tension and rough sound. Chord tension is an important factor, which affects one of the two important axes of emotion plain, arousal. We devise a method for directly extracting the chord tension from the signal, while bypassing the premature chord recognition and transcription system. The next feature we propose is rough sound. Rough sound is the noisy components in the song, like drums or distorted electric guitars. We propose a computationally competitive, but well-performing rough sound extraction method compared to the existing music source separation technology.

The newly developed AMC system is evaluated with the combinations of proposed features using the verified MIREX datasets. With the careful exploration and optimization, the proposed AMC system outperforms the whole submitted systems of recent two years' MIREX.

KAIST

Table of Contents

KAIST

List of Tables

KAIST

List of Figures

KAIST

KAIST

List of Abbreviations

AMC	Audio Music Mood Classification
GUI	Graphical User Interface
MFCC	Mel-Frequency Cepstral Coefficients
MARSYAS	Music Analysis, Retrieval and Synthesis for Audio Signals
MIREX	Music Information Retrieval Evaluation eXchange
SVM	Support Vector Machine
DCT	Discrete Cosine Transform
PCP	Pitch Class Profiles
DFT	Discrete Fourier Transform
SFM	Spectral Flatness Measure
BPM	Beat Per Minute
CQT	Constant-Q Transform
STFT	Short Time Fourier Transform
RBF	Radial Basis Function
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis
LPP	Locality Preserving Projections

KAIST

I Introduction

1.1 Motivation

The ability to efficiently retrieve data from the mass storage of music database has become a crucial issue with the rapid growth of related research areas, such as digital signal processing, machine learning, and information retrieval [1]. Traditionally, people can search music only by its title, name of artist, lyric, and so on. However, sometimes queries cannot be in the form of these conventional representations. This paper concerns one of these alternative descriptions of music, which can be called 'mood'. We assume that the users want to listen to some songs which are appropriate in their mood. To satisfy their needs, it is very important for the system to automatically classify audio music by the mood. Actually, it is impossible for the music experts or common users to manually put mood tags on massive music database.

Therefore, many audio music mood classification (AMC) systems, which can categorize music automatically, have been developed so far. Furthermore, novel music exploration services are emerging, which are based on higher level of music description as their interface with the users. The Mood Cloud system, for example, provides mood-based Graphical User Interfaces (GUI) for the users to find songs more intuitively and efficiently [2]. Assume that the users want to listen to some cheerful songs in the gloomy morning. They need to recall the melody of appropriate songs and then try to figure out their titles or the name of artists who made them. In order to make their playlist long enough for their breakfast and quick shower, they need to spend at least couples of minutes for creating the playlist itself. However, with the help of alternative representations about the songs, the users can simply click the keyword of the music exploration system, ‘cheerful’ for instance, and then listen to cheerful songs in the automatically created playlist. Figure 1 gives us the pictorial example of GUI in

KAIST

Mood Cloud system.

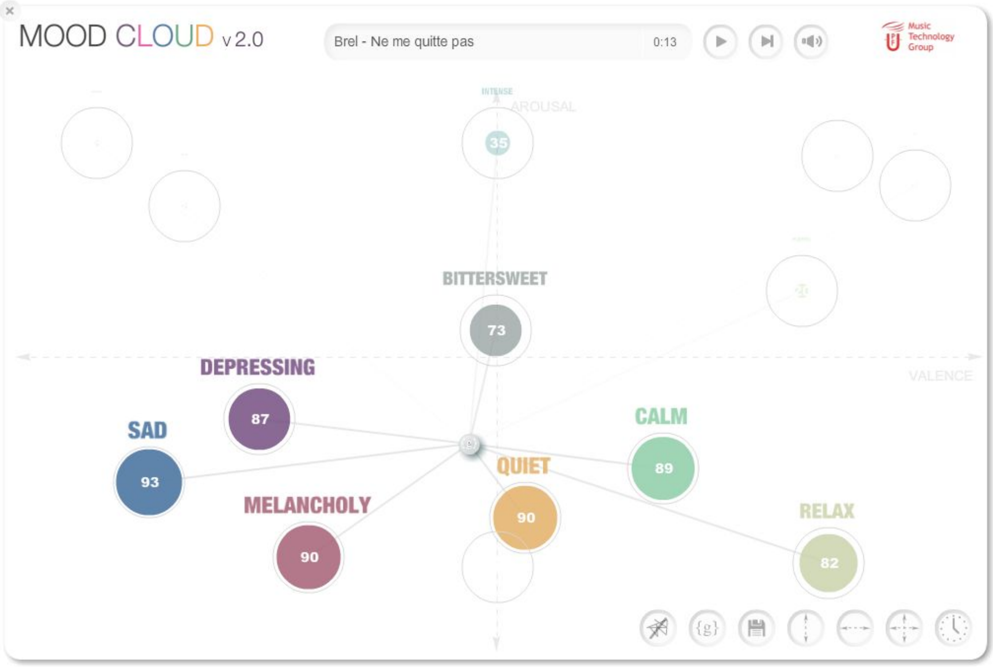


Figure 1. GUI example of Mood Cloud system [2]

KAIST

Although the existing AMC systems work reasonably well, they usually use low-level features such as Mel-Frequency Cepstral Coefficients (MFCC) and other spectral features which are not enough to deal with very structured general music. On the contrary, higher-level features, such as chord, rhythm, and instrumentation, are more likely to express mood information of music. Figure 2 shows an example of hierarchical structure of features which can be used in AMC systems [3].

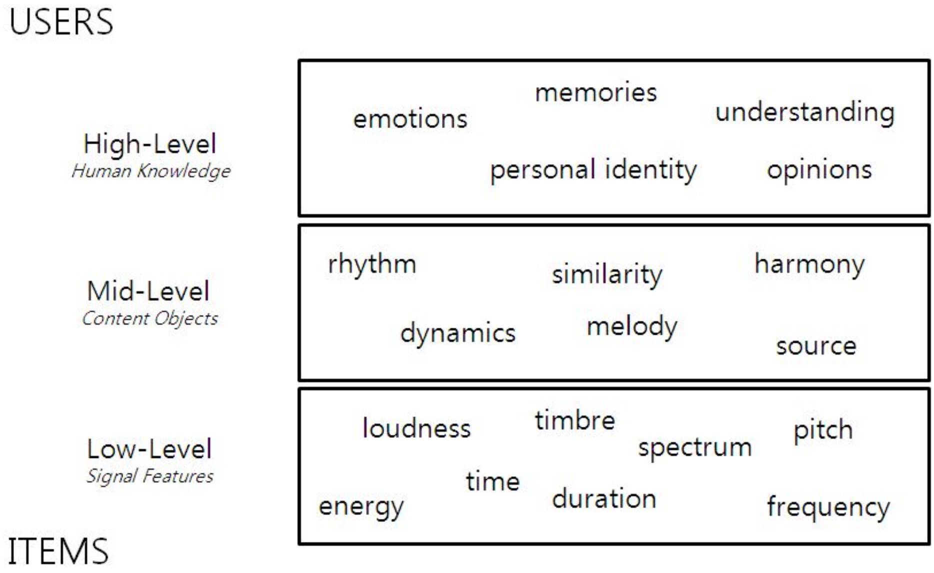


Figure 2 Hierarchical structure of features used in AMC systems [3].

1.2 Idea

KAIST

In this work, we aim at exploiting mid-level music features into the AMC system. The most plausible way to do this is to extract those features and use them as symbolic forms in classification system. However, the relatively low performance of those mid-level feature extraction systems can be another cause of degradation of total performance of AMC system. In this work, we try to find a way to avoid the degradation of total performance, yet effectively extracting mid-level feature.

The firstly proposed feature measures chord tension directly. The chord tension literally affects how tense a song is, so we believe that it is relevant to arousal axis of emotion space very much [4]. It is true that we can easily measure the tension of a given symbolically represented chord, *CM7* for example, if we can exactly guess from the signal what the chord is. However, relatively poor performance of chord extraction methods, under 70% at most even with subset of all possible chords [5], we need to use another method for introducing the concept of tension into the AMC system. In this thesis, therefore, the chord tension extraction method, which does not involve existing chord recognition tools, is devised to avoid the error of chord recognition itself. Based on musicology, we define the tension of the chord as its distance from the tonic chord [6]. We also define the distance between chords as the degree of harmonic coincidence between the given two chords. To measure the distance, we extract the harmonic component from the frequency spectrum. K-means clustering follows to find the chord clusters from the processed signals, and then we compare the cluster means, as the representative of each frame, with the tonic chord of the song clip in Euclidean distance. By summing up the distances, we can get the total tension of the song, approximately. The proposed chord tension feature does improve the performance of AMC system in spite of its imperfect ability to recognize chords from signals.

The second feature is designed to extracts some noisy components of the input signal, which are spectrally spread sounds, such as drums and distorted electric guitar sound. This feature can work for measuring the degree of roughness or the portion of drums in the song. For example, the value of second feature will be lower with the songs which are acoustically soft, compared with those have strong drums and noisy sounds. Another merit of this feature is that the AMC system can capture those highly emotion-related components without complex drum source separation technique or rhythm feature extraction tools. To get those components, we use two successive simple filters for removing harmonics of the input signal, which can be regarded as impulses along the spectral axis. After summing those processed signal, we can get the feature which approximately shows the portion and behavior of rough sound components in the songs.

KAIST

1.3 Thesis Contributions

The contributions of this thesis are as follows:

- This thesis proposes the definition of chord tension as a feature of AMC system, which is not based on the symbolic representation of chord, but the raw signal directly.
- This thesis proposes the method of extracting the chord tension feature and verified the procedure empirically.
- This thesis proposes the definition of rough sound as a feature of AMC system.
- This thesis proposes the method of extracting rough sound and verified the procedure empirically, which has superiority in its complexity.
- This thesis finally improves classification performance of AMC system with well-known music database by using:
  - the abovementioned proposed mid-level features by this thesis,
  - the carefully chosen parameters through classifier optimization,
  - and the already existing low level features of MARSYAS (Music Analysis, Retrieval and Synthesis for Audio Signals).

KAIST

1.4 Thesis Overview

The rest of the paper is organized as follows: Chapter 2 describes background of this study and the related works. The proposed two novel mid-level music features are presented in Chapter 3. Chapter 4 shows experimental environments and results. Finally, we summarize our work and present future directions in Chapter 5.

II Background and Related Works

2.1 MIREX Framework

2.1.1 Mood Categories

We use the five mood categories which MIREX (Music Information Retrieval Evaluation eXchange) defined [7]. The mood clusters are made of carefully chosen keywords, which are compact representatives of various definitions of human emotion, and yet basing on widely believed relationship between the mood and music [8].

Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**
Rowdy	Amiable/	Literate	Witty	Volatile
Rousing	Good natured	Wistful	Humorous	Fiery
Confident	Sweet	Bittersweet	Whimsical	Visceral
Boisterous	Fun	Autumnal	Wry	Aggressive
Passionate	Rollicking	Brooding	Campy	Tense/anxious
	Cheerful	Poignant	Quirky	Intense
			Silly	

Table 1. Five mood clusters used in the AMC task [9]

2.1.2 Ground Truth Sets

We use 132 exemplar songs for the development of our AMC system which MIREX offers. The audio set is pre-labeled with those five mood clusters according to their metadata. To make sure the mood labels are correct, this audio collection was validated by human subjects: the audio clips, whose mood category assignments reach agreements among two out of three human assessors, were chosen as a ground truth set.

ARTIST	TITLE	CLUSTER
U2	Where the Streets Have No Name	1
blink-182	What's My Age Again?	1
Bryan Adams	Summer of '69	1
Lynyrd Skynyrd	Gimme Three Steps	1
Foreigner	Double Vision	1
Green Day	Basket Case	1
Cyndi Lauper	Girls Just Want to Have Fun	2
Neil Sedaka	Calendar Girl	2
Stevie Wonder	You Are the Sunshine of My Life	2
Spice Girls	Wannabe	2
The Bangles	Walk Like an Egyptian	2
ABBA	Take a Chance on Me	2
America	Sister Golden Hair	2
The Everly Brothers	Problems	2
Culture Club	I'll Tumble 4 Ya	2
Creedence Clearwater Revival	Down on the Corner	2
The Everly Brothers	Claudette	2
Simon & Garfunkel	The Boxer	3
The Bee Gees	How Can You Mend a Broken Heart?	3
Coldplay	Yellow	3
Simon & Garfunkel	The Only Living Boy in New York	3
Belle & Sebastian	The Fox in the Snow	3
The Verve	The Drugs Don't Work	3
The Beatles	Something	3
Neil Young	Old Man	3
The Moody Blues	Nights in White Satin	3
Bruce Springsteen	My Hometown	3
Radiohead	Lucky	3
Fleetwood Mac	Landslide	3
Radiohead	Karma Police	3
Billy Joel	Just the Way You Are	3
Roy Orbison	It's Over	3
Rod Stewart	Gasoline Alley	3
R.E.M.	Everybody Hurts	3
Crowded House	Don't Dream It's Over	3
Roy Orbison	Crying	3
Radiohead	Creep	3
Procol Harum	A Whiter Shade of Pale	3
Stephen Malkmus	Troubbble	4
The Beatles	Taxman	4
Soft Cell	Tainted Love	4
Talking Heads	Swamp	4
Violent Femmes	Blister in the Sun	4
Violent Femmes	Add It Up	4
Nirvana	Aneurysm	5
Alice in Chains	Would?	5
Nirvana	Smells Like Teen Spirit	5
Metallica	Master of Puppets	5
Faith No More	Epic	5
Rammstein	Du Hast	5

Table 2. List of exemplar songs. Only 51 out of 132 songs are represented.

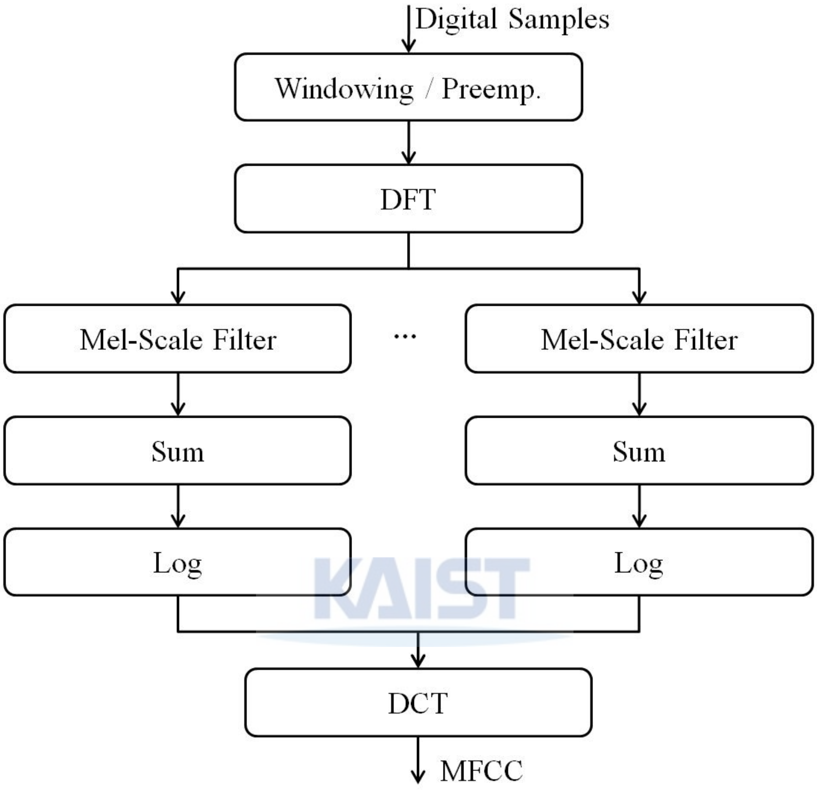
The exemplar dataset is not satisfying to guarantee the performance of the AMC system since it is not evenly balanced. Moreover, it is not enough in their amount. The exemplar dataset is only for reference, so that MIREX does not guarantee that the AMC system, which works well with the exemplar dataset, also does with the 600 ground truth songs, which are actually used in MIREX AMC task.

Likewise, to keep the MIREX contest fair enough, the committee introduced small exemplar set for reference, yet maintains both the list and files of the whole ground truth dataset in secret. However, they run the submitted systems with the ground truth dataset and report the results to the applicants. After finalizing our features and system with the 132 songs, we also check them with the 600 ground truth dataset by submitting our system to the MIREX committee. Our system is also fairly examined with the actual ground truth dataset and the classification results are drawn by the MIREX committee.

KAIST

2.2 Audio Music Mood Reference System

2.2.1 MARSYAS



KAIST

Figure 3. Block diagram of procedure to get MFCC from digital samples.

The open-source music classification solution, MARSYAS, is very famous and widely referred not only for its robust performance, but for its usability [10, 11]. It marked 61.5% at MIREX 2007, 58.2% at MIREX 2008 in the AMC accuracy. This system learns the relationship between music and mood through Support Vector Machine (SVM), and uses temporally abstracted statistics of MFCC and several timbre features.

MFCC is a well-known timbre estimation feature which has been widely used in speech recognition systems [12]. Figure 3 describes the procedure of MFCC extraction. Given a set of linear spectral components, MFCC firstly sums up the mel-scaled filtered output to reflect the spectral characteristics of the human auditory system. Then, it takes logarithm and transforms them with Discrete Cosine Transform (DCT).

MARSYAS also works with some basic spectral features. Spectral centroid decides whether the spectral components of a given frame are distributed in the low or high frequency. Spectral roll-off point is the point where the accumulated value of spectral components reaches 85% of the total spectral energy from the lowest frequency bin to the highest one. It also shows spectral distribution of a frame. Finally, spectral fluctuation, or simply flux, represents the temporal variation of spectral components.

Figure 4 shows the temporal approximation procedure of MARSYAS. The features of MARSYAS, which are drawn in frame-by-frame manner, are put together and make a single feature vector. To make this procedure more meaningful, MARSYAS takes 43 feature vectors, which are one second long in the MIREX experimental environment, and then calculates the sample mean and standard deviation. To capture the temporal variation of the features, MARSYAS takes next 43 feature vectors and get the statistics again, 42 of them are the same with the previous calculation, by the way. MARSYAS calls this one second long sliding windowed manner of feature abstraction procedure ‘texture window’. Finally, MARSYAS gets final sample mean and standard deviation of the texture windowed means and standard variations to approximate the feature vectors of every frame into a single feature vector. In this thesis, we follow this texture window and single feature vector approximation schemes identi-

KAIST

cally for our proposed features.

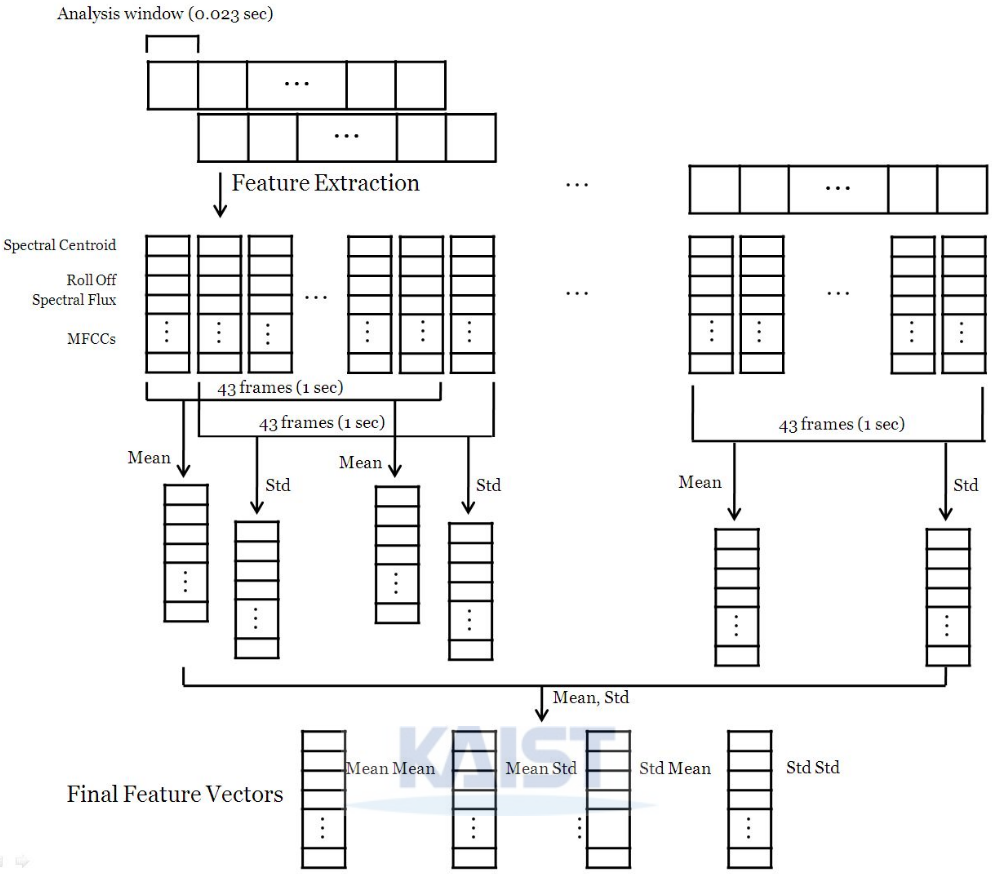


Figure 4. Temporal approximation procedure of MARSYAS using textual window.

Figure 5 shows the whole procedure of MARSYS' train-and-prediction system, which is very conventional form of classification. MARSYAS pursues universal goodness in the various classification tasks and simplicity in its structure. Therefore, it works well in almost train-and-test tasks of MIREX, such as artist classification, genre classification, and music mood classification. However, the AMC task, which the thesis is focusing, does need more mid-level features, where the emotional latent information of the signal is reflected, in addition to the universally working low-level features of MARSYAS.

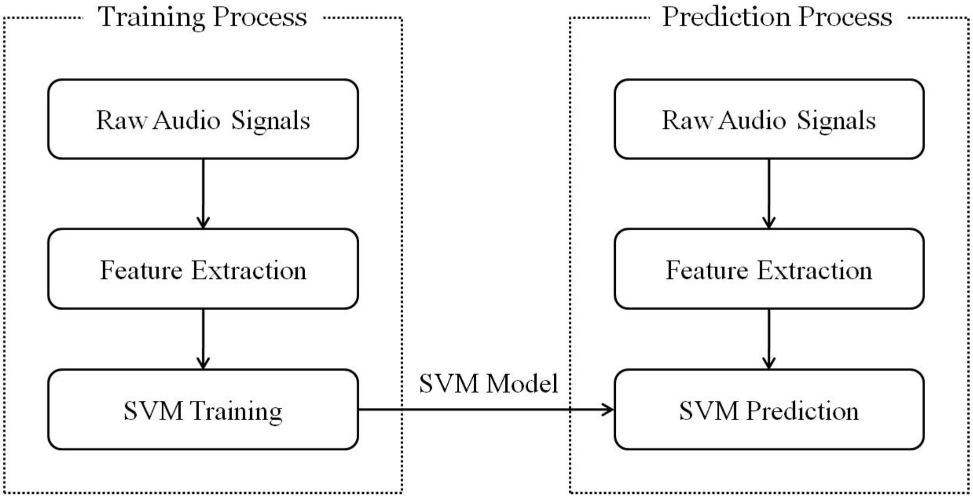


Figure 5. Whole procedure of MARSYAS' train and prediction system

KAIST

2.3 Audio Music Mood Features

2.3.1 Low Level Music Features

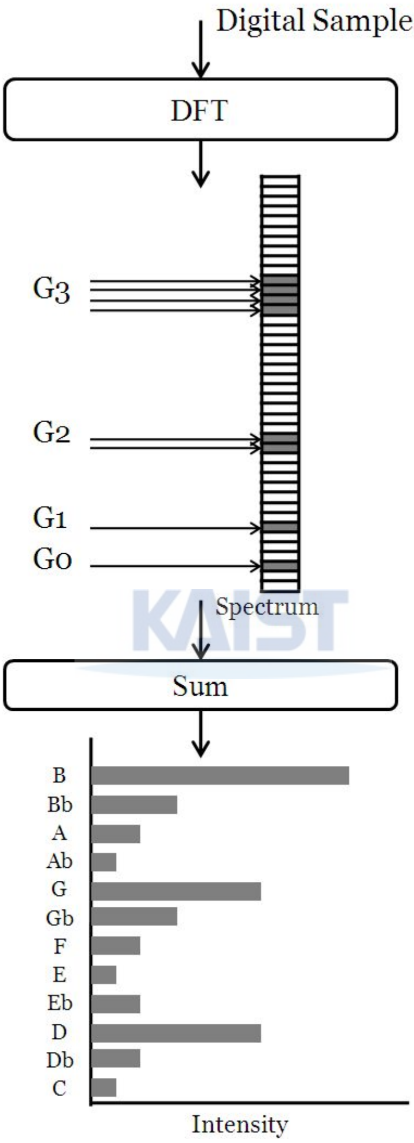
The process which selects features from the signal is very important in train-and-test system. Many low level music features have been developed in the music information retrieval fields. In this chapter, we will introduce a few more general features aside from the MARSYAS features.

Chroma, which is also called Pitch Class Profiles (PCP), is a low level feature which extracts harmonic components from the frequency domain signal [13, 14]. This feature extracts the only frequency components which are lying in the pre-defined musical pitch frequency. Then, it sums up the components whose distances are octave long to get the pitched frequency component regardless of its octave. For instance, in the chroma extraction process, the nearest frequency bins of the input spectrum to the pre-defined pitch frequencies, such as 220Hz (*A3*), 233Hz (*Bb3*), 247Hz (*B3*), 261.6Hz (*C4*), and so on, are collected for the seeds. After that, frequency bins corresponding to 220Hz, 440Hz, 880Hz and their series are summed to eliminate the octave effect. Another AMC system adapts this feature [15], and most of the chord recognition processes use this as a preprocessing step [16, 17, 18]. However, when we used chroma as a feature directly in the Marsyas system, it did not help improve the performance of classification accuracy. Actually, chroma needs additional post-process to catch the harmonic characteristics which we want to find.

Figure 6 shows pictorial representation of chroma extraction process with Discrete Fourier Transformed digital samples [13]. Figure 7 shows comparative results of two chords, *C* and *Cdim7*, played with flute. Compare to the high resolution of Discrete Fourier Transform (DFT) results, chroma does not seem to show harmonics components of the signal, but it sums up them into the same octave group. As for the log-scaled mel-frequency cepstrum result, it provides rougher representation of the spectrum which can be regarded as an envelope of the spectrum.

KAIST

Spectral Flatness Measure (SFM) is another famous feature to catch a flatness level of the frame in the spectral representation [19]. The flatness is useful to decide whether the spectral distribution of a given frame is noise-like or not, because noisy component tends to have flatter spectrum than harmonic component. However, this feature also needs to be improved since it can be confused with some frames where the harmonic and inharmonic components are coincidently playing. Likewise, in the polyphonic music, many instruments are mixed with one another, so that it cannot be guaranteed that the pure harmonic part and pure drum part exist in a song. If we want to know more exact property of noisy components of the song, we need to extract or separate them first and then process them with spectral features like SFM.



KAIST

Figure 6. Chroma extraction process

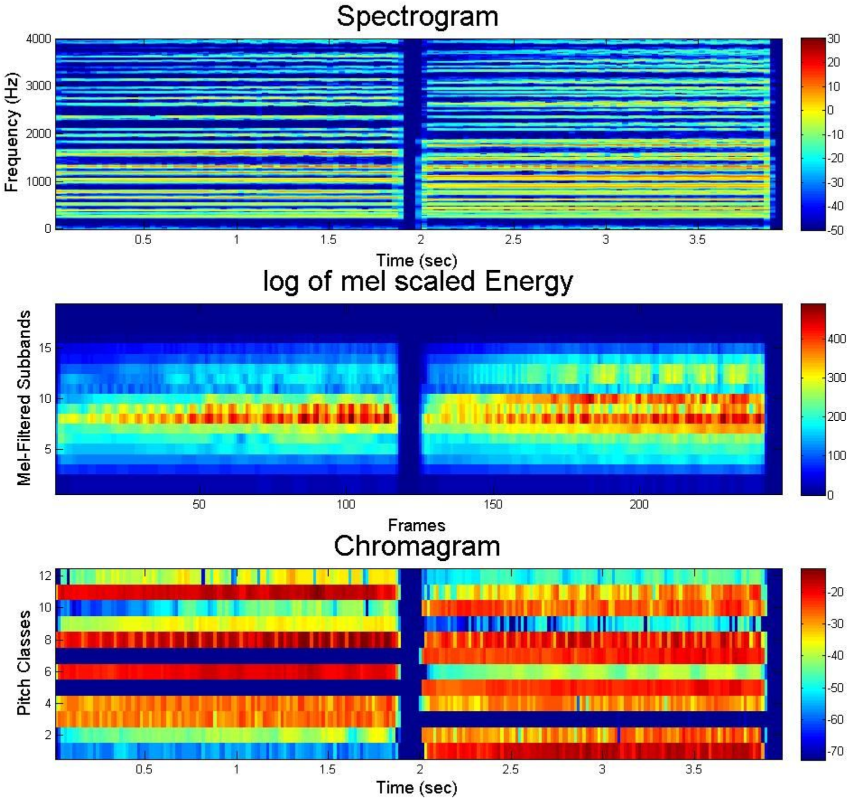


Figure 7. Comparative time-frequency representations of two successive chords, *C* and *Cdim7*, played with flute. DFT spectrogram (top), log of mel-scaled energy (middle), and chromagram (bottom).

KAIST

2.3.2 Mid-level Music Features

There have been several trials of adding a mid-level feature in the form of symbols for improving performance of AMC system. For instance, [20] improved the accuracy of emotional valence prediction by using chord histogram which is devised for representing distribution of a set of estimated chords. Even though this work was done outside of MIREX framework, this study showed promising results for us that chord-related feature can improve the performance of AMC system. On the contrary, the chord histogram feature is effective mainly for predicting the emotional valence, which is a continuous representation about the degree of brightness or happiness of the feeling. We assume that the chord sets, which [20] defined, are not enough to take the harmonic arousal information into account. Our chord tension feature, by the way, desires to predict the tension of a given song, which can be viewed as an emotional arousal in other words.

Although we can concede that the chord-related features are plausible for AMC system, finding exact chord information from the complex commercial music is not an easy task. In the 2008's MIREX audio chord detection task, the averaged performance of chord detection accuracy was under 70% at best. Furthermore, as we explain in the following sections, symbolic chord itself does not provide tension information directly. In order to decide how tense a chord is, we need to extract quantitative tension information from the symbolic chord or from the signal directly.

Another famous mid-level feature is tempo of the song. Tempo is very effective feature to convey composer or performer's moods to the audience since it decides the speed of the song. For example, people often prefer to listen to faster songs than slower ones when they are driving fast. Similarly, when people are depressed, fast song makes them energetic. On the other hands, when people are restless, slow song makes them calm down and feel comfortable. Aside from the intuitively clear effectiveness of tempo as an AMC feature, finding tempo of the song is another big thing to work with. At first, it is hard to decisively define the tempo of songs in many cases since they usually contain both the frequently occurring instruments and sparsely doing ones. Therefore, people often cannot assure a song's tempo in Beat Per Minute (BPM) when the song can contain both 120 BPM hi-hat and 60 BPM snare drum. Coupled with this perceptual confusion, finding out the massive numbers of onsets in

KAIST

the signal and tracking the time-varying beats are well-known problems to be attacked in automatic tempo detection task.

III Proposed Mid-level Music Features

In this thesis, two mid-level music features are proposed: the chord tension feature and rough sound feature. This chapter firstly considers why those mid-level music features are promising to improve the performance of AMC system. Next, the proposed algorithms are analyzed with the intermediate product resulted from each step of the algorithms. Finally, we evaluate how powerful those features are for capturing the desired mid-level music characteristics.

3.1 Harmonic Feature

3.1.1 Chord Tension

Chord can be defined as a set of simultaneously playing notes regardless of their octave and specific instrument which plays the notes. *C* chord for example, consists of three notes, *C*, *E*, and *G*. By the definition of chord above, we also call all the combination of a variety of sets of octave-differentiated notes, *C4*, *G3* and *E6* for instance, as *C* chord as long as they are made of those three notes.

KAIST

Another important thing about chord in human perception is that people are more likely to assume a relatively long time period as a chord section even though there exist some out-of-chord passing notes. For instance, it is more plausible to consider a chord as a time interval than a moment, when the accompanying notes are played in broken-manner, not simultaneously.

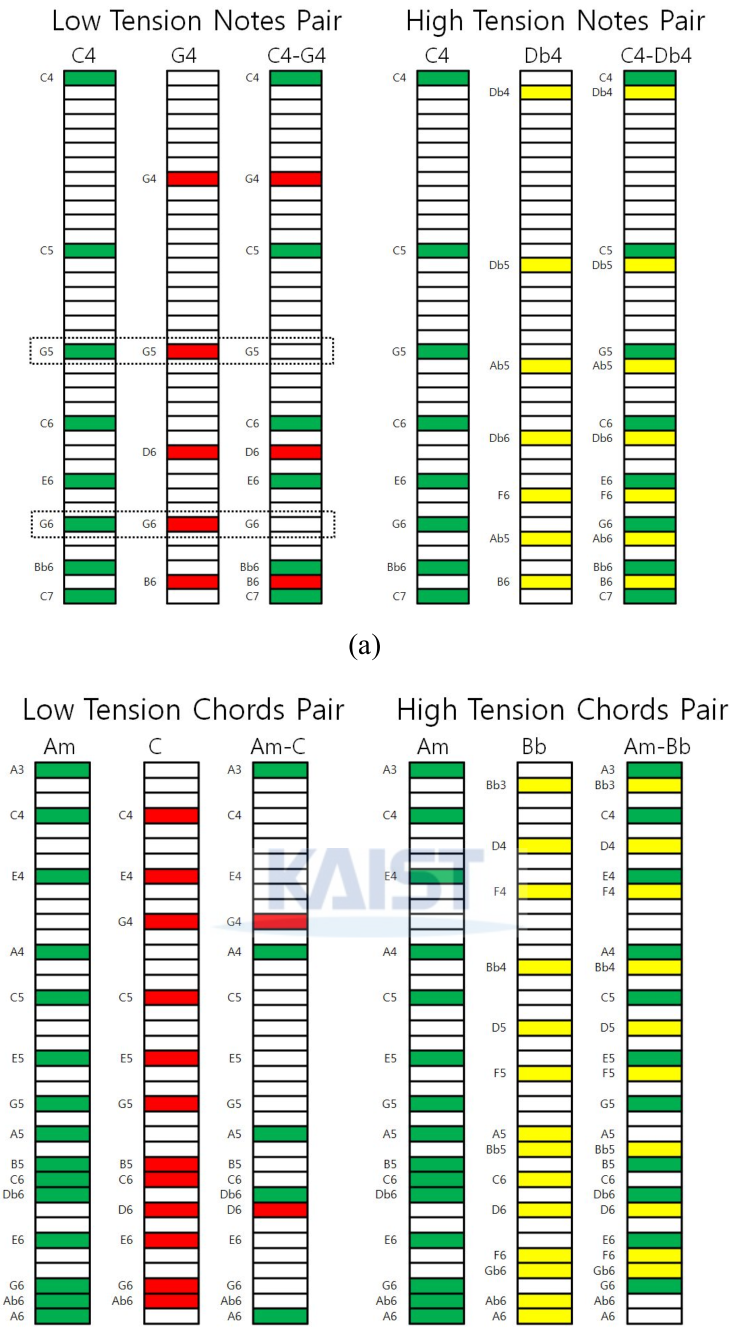
The symbolized chord information can plays a big role in the music classification system as a mid-level feature, since chords can provide us harmonious structure of the songs. It is very reasonable that minor chords convey somewhat sad or gloomy mood compared to major chords. For example, the estimated major or minor chords were used as a feature to improve the performance of the emotional valence prediction [20].

On the other hands, some chords can generate uncomfortable feeling in the given key, so that they increase the tension of the whole song: *Db* or *Gb* chord in the *C* key. Similarly, chord itself can have its own tension information when there are tension notes in addition to common triad: *C7* which is made of additional *Bb* note to the original triad of *C* chord. Human auditory system can recognize those tensions not only between key and a chord, but lying in intra chord. However, using the symbolized chords as a feature for music information retrieval system has lots of difficulties because of the performance limitation of the chord recognition technology.

What we consider in this thesis is the chord tension of a song. Chord tension means the tension of the chord and it affects to the tension or arousal aspects of the mood. We cannot fully recognize the chord tension using the traditional automatic chord recognition technology, because it barely distinguishes 24 possible major and minor chords and some tensional extensions. In order to overcome the limitation of chord recognition performance and to get the tension information more safely, we decide not to try to know the exact name of the chord, but to distinguish them with their quantitative tension values.

KAIST

Based on musicology, we define the distance between chords or notes as the degree of harmonic coincidence between them. Moreover, we also define the tension of a given chord as its distance from the tonic chord (key). Figure 8 shows pictorial example of distance between two notes and two chords. Figure 8 (a) gives us the fact that the harmonics of the two single notes *C* and *G* coincide more than that of *C* and *Db*. Therefore, we can conclude that *C* and *G* are less tense than *C* and *Db*. It agrees with the musicological truth and human assessment tests about the tension between notes [6]. Similarly, Figure 8 (b) also adapts the same principle about tension between two chords: the level of harmonic coincidence. The two chords *Am* and *C* coincide more in their harmonics than *Am* and *Bb*, which are known for tenser pair.



KAIST

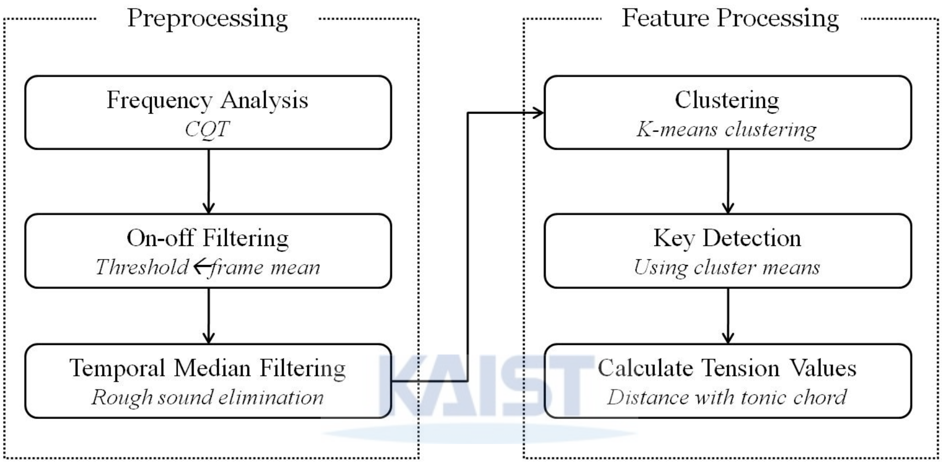
(a)

(b)

Figure 8. Harmonic coincidence of two notes (a) and two chords (b)

3.1.2 Proposed Method

The approach for computing the chord tension in the thesis follows the process shown in Figure 9. First, there is a spectral analysis, followed by the extraction of harmonic components where the timbral characteristics are also removed. Then, we eliminate rough sound components with the help of their temporal property. Next, we calculate the distance between the representative chord of the frame and the tonic chord of the song after allocating each frame to appropriate chord cluster.



KAIST

Figure 9. Chord tension extraction process

We use Constant-Q Transform (CQT) [21] to analyze the spectral components from the raw audio signal. We are interested in pitch-related frequencies, but ordinary DFT carries the uninteresting frequency bins as well, because it divides the frequency axis in an equal space. Figure 10 shows an example of CQT spectrogram. We need to remove the timbral characteristics and rough sound components from this spectrogram in order to emphasize the harmonic components.

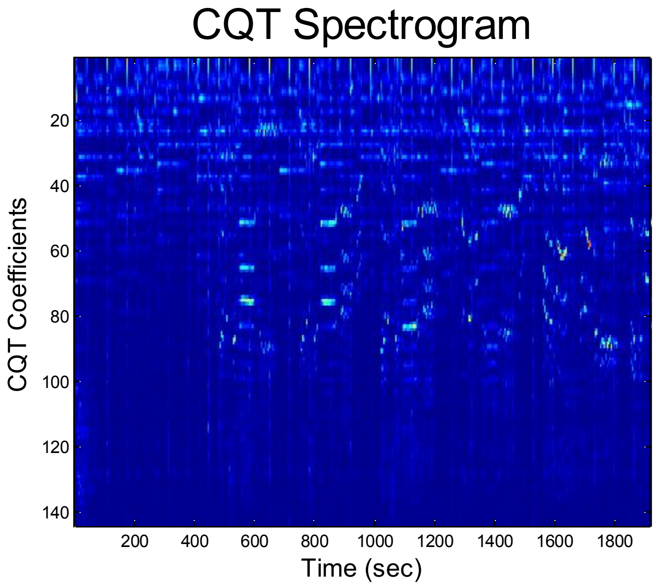


Figure 10. An example of CQT spectrogram

To get rid of rough components and timbral characteristics, we turn on the only frequency bins whose energy is big enough to be regarded as harmonics. By letting all the turned-on bins have the value one, we could also eliminate timbral characteristics of the harmonics which can harm the tension measurement. Zero is assigned to all the other turned-off bins, on the other hands. We call this process the on-off filtering. Figure 11 shows on-off filtered spectrogram where red bins mean turned-on while blue bins mean turned-off. We can find that there are some noisy bins which obstruct distinguishing harmonics of the input signal.

KAIST

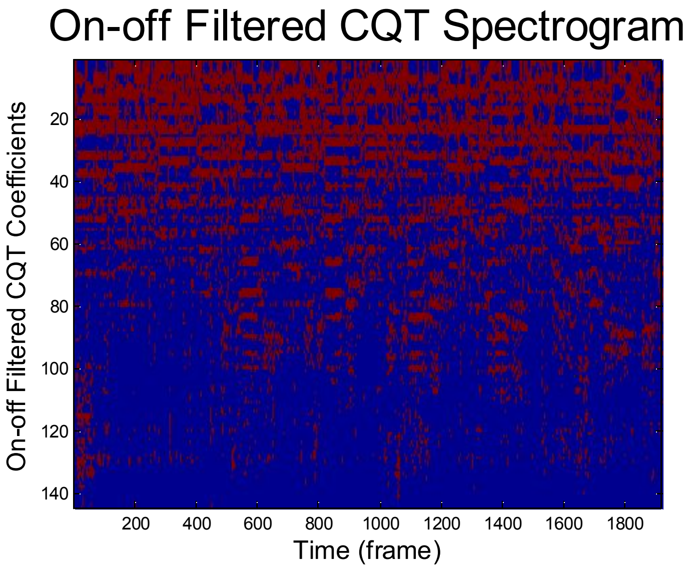


Figure 11. An example of on-off filtered CQT spectrogram

KAIST

Then, we median-filter the on-off filtered frames temporally to eliminate drums or needless noise. The temporal median filtering can be regarded as a temporal noise reduction tool which is devoted for wiping out impulsive sound, like drums. The harmonious instruments, on the contrary, are apt to be continuously long enough not to be eliminated by temporal median filtering. From the Figure 12, we can identify the harmonic components are remained well while the noise components are removed, after on-off and temporal median filtering.

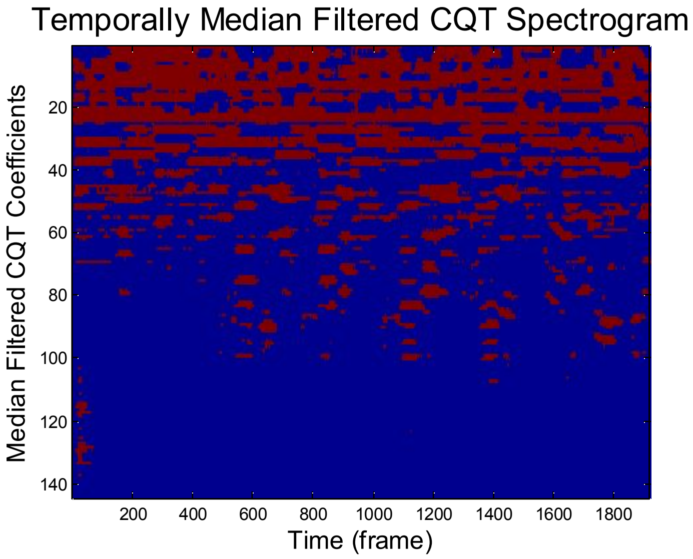


Figure 12. An example of temporal median filtering after on-off filtering to the CQT spectrogram

KAIST

After getting the harmonic component, we need to cluster the frames based on the chords they are making. We does not use the supervised learning technique for clustering the frames, because the accuracy of supervised chord recognition technique is not satisfying aside from the fact that current chord recognition techniques do not cover all possible chords. Unsupervised learning techniques, however, are not needed to construct enormous size of chord template database for training. Furthermore, they can distinguish chords more specifically with less assumed numbers of clusters. In addition to that, we do not need to get the exact chord name, but just want to group the frames based on chord tension, so we choose k-means clustering algorithm for grouping the on-off and median filtered frames. K-means clustering is a simple, but widely used clustering algorithm for its simplicity and relatively good performance. We pick up the value ten for the number of clusters, K, because the number of chords in 30 second excerpt of a song usually does not exceed ten. Moreover, we also check that which value of K results best in AMC performance. Table 3 shows the classification accuracies for three different values of K where we can also see that the value ten performs best.

Num. of Clusters	**6**	**8**	**10**
Classification Accuracy	52.50%	50.94%	52.98%

Table 3. Classification accuracies for different numbers of clusters.

Figure 13 and 14 represent cluster means and allocation of each frame, respectively. The cluster labeling is done manually after k-means clustering in Figure 13. Passing notes does interfere clustering by separating the same chord section into different clusters, but we can see that many harmonics of the different clusters are actually overlapping much if they are the same chord. Figure 14 tells us that each frame is allocated well to the cluster, which represents its original chord.

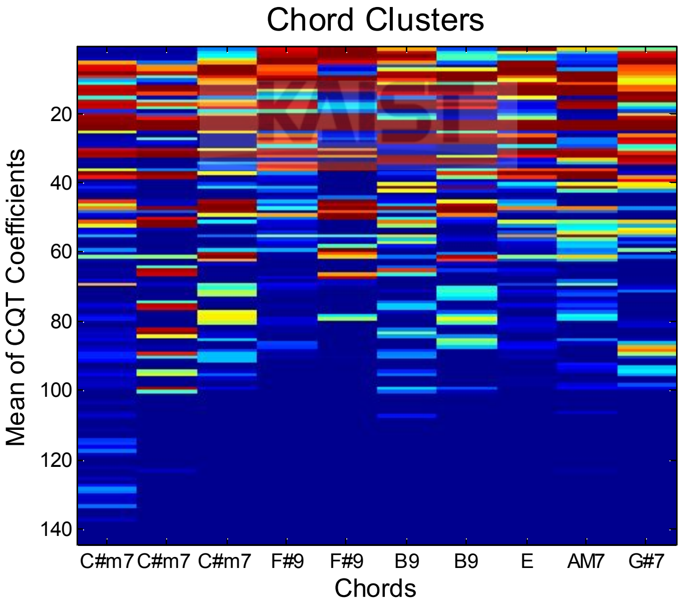


Figure 13. Manually labeled cluster means.

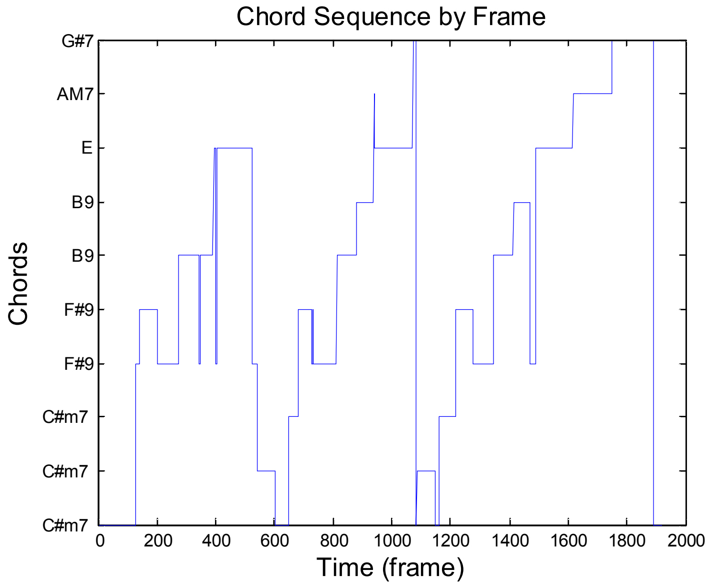


Figure 14. Allocation of each frame to a chord cluster

After clustering, we compare the cluster means, as the representative of each frame, with the tonic chord of the song clip. We use Euclidean distance for measuring the difference. Figure 15 shows frame-by-frame tension values that also reflect perceptually and musicologically verified actual tension well. By summing up the distances, we can earn the total tension of the song, approximately.

KAIST

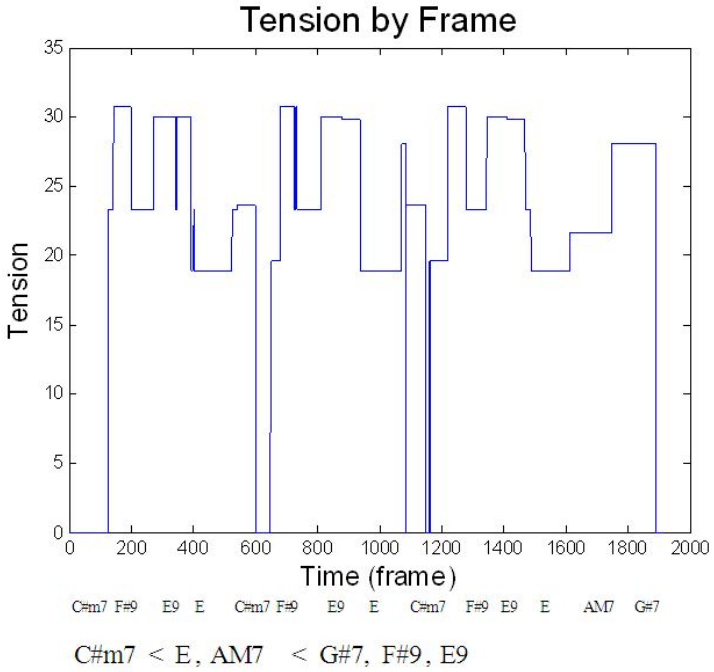


Figure 15. Frame by frame tension values and actual chord tension

In order to find the keys of each input songs, we assume that all of the cluster means can be regarded as a tonic chord. After iteratively choosing one of the cluster means as a candidate tonic chord, we calculate tensions between the candidate tonic chord and the other cluster means. Then, we select the one with the lowest tension with the other cluster means as the winner, based on the intuitively clear assumption that the distance between the real tonic chord and all the other chords will be the lowest of all candidate tonic chords. Suppose that there are six common chords in a song with *C* key: *C*, *G*, *F*, *Dm*, *Em*, and *Am*. If we choose *C* chord as the tonic chord properly, we can see that the other chords *G*, *F*, *Dm*, *Em*, and *Am* are very common and are not tense much. However, if we select *G* chord as the tonic chord, *F* and *Dm* chords become uncommon and are tenser than that case of *C* chord.

KAIST

To summarize, we can conclude that the chord clustering result and the obtained chord tension represent the tension of the chord quite well, except very noisy frames.

3.2 Rough Sound Feature

3.2.1 Property of rough sounds

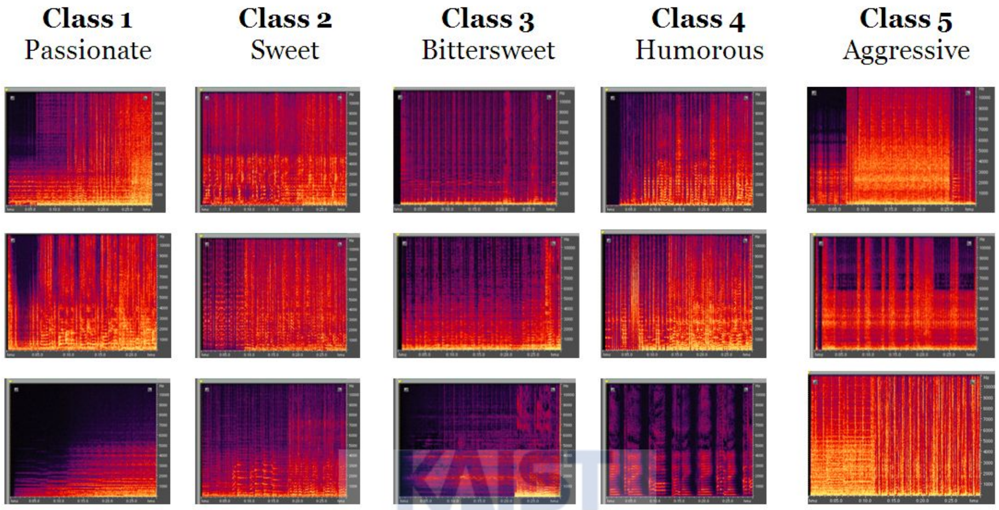
Rough sound plays another important role in conveying mood of the song. In this thesis, we define the term, *rough sound*, as noisy and dissonant sound components which usually do not have much harmonics in its spectral aspect. They tend to be flatter in their spectral shape compared to the harmonious components, so they are usually used for controlling the amount of inharmonic excitation in the song through their degree of loudness and repetition. For example, as the percussive sound is repeated dynamically, the arousal aspect of mood increases. When the tempo of music is faster, both the valence and arousal aspect of the mood is higher, too.

The most common rough sound components in music are percussive or rhythmic instruments. We concede that there are some exceptions, like timpani, bells and triangle, indeed carry their own harmonics in their sounds while they are usually grouped into percussive instruments. However, in most cases, the conventional drum sets for example, percussive instruments are more apt to be perceived as rough sound since their inharmonious characteristics.

KAIST

Likewise, if we take the inharmoniousness of the sound components into consideration, we need to measure the amount of roughness of a sound component even though it is partly harmonious, but also inharmonious. In rock music, for instance, musicians depend significantly on electric guitars with artificial distortion which adds a kind of noise floor to the harmonics of guitar strings. In that case, the consonance of the electric guitar sound can be harmed, and then the roughness of the sound grows.

Figure 16 shows the examples of spectrogram per each mood category. Class 5 usually consists of the heavy metal songs which convey aggressive and fierce mood with strong drum and distorted electric guitar sounds. We can see that their spectrums are full of not only strong drum sounds, but the noisy harmonious components from electric guitar. On the other hands, class 3 consists of ballads and soft songs which express bittersweet and poignant emotion with relatively weak drum and pure sounded instruments.

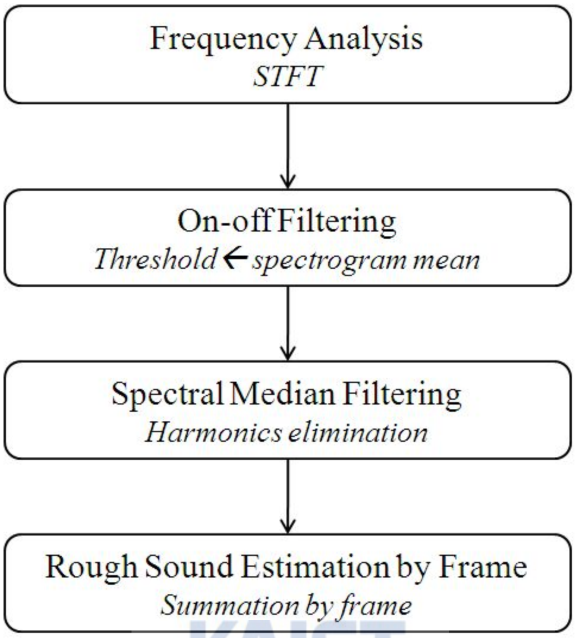


KAIST

Figure 16. Examples of spectrogram per each mood category.

We can simply imagine that measuring the amount of rough sounds in the multi-instrumental music will be easy if we have the unmixed original sources. Otherwise, it could be also plausible if we can extract the rough sound sources from the mixed one. However, the music source separation is very difficult because of the lack of the number of mixtures and dynamics of mixing environments. We can simply adapt the current drum source separation technique [22], but it is computationally very complex and time consuming with its unsatisfying separation performance, because it should be run on all hundreds of input music.

3.2.2 Proposed Method


Frequency Analysis
STFT
On-off Filtering
Threshold ← spectrogram mean
Spectral Median Filtering
Harmonics elimination
Rough Sound Estimation by Frame
Summation by frame

KAIST

Figure 17. Block diagram of rough sound extraction procedure

This section explains the proposed lightweight rough sound estimation method. The approach for distinguishing the rough sound follows the process shown in Figure 17. First, there is a spectral analysis using DFT instead of CQT, which was used in chord tension extraction, since the fine resolution of low frequency spectrum is not required in rough sound extraction. Figure 18 shows an example Short Time Fourier Transform (STFT) spectrogram. We need to remove the timbral characteristics and harmonic components from this spectrogram in order to emphasize the drum and noisy components.

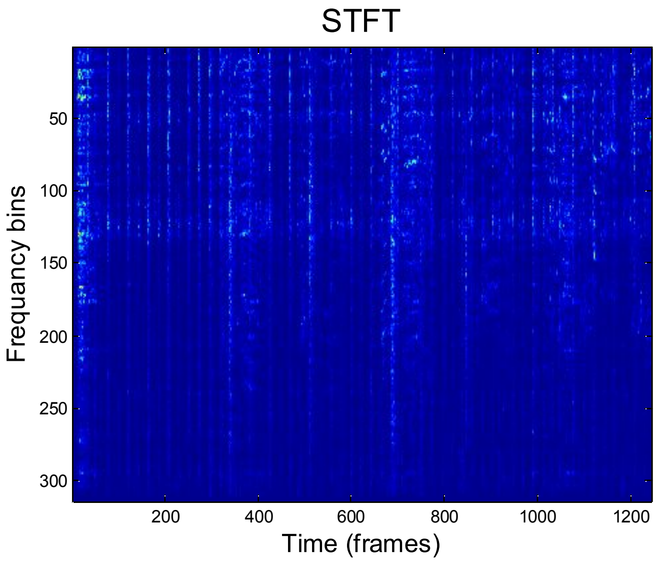


Figure 18. An example of STFT spectrogram

KAIST

On-off filtering, which is based on the total sample mean of the spectrogram as its threshold, follows. Similarly to the purpose of on-off filtering of chord tension extraction process, the on-off filtering phase in this step aims at removing the timbral characteristics. Figure 19 shows on-off filtered spectrogram where red bins mean turned-on while blue bins mean turned-off. We can find that harmonics structures are remained yet, which are not the part of the rough sound.

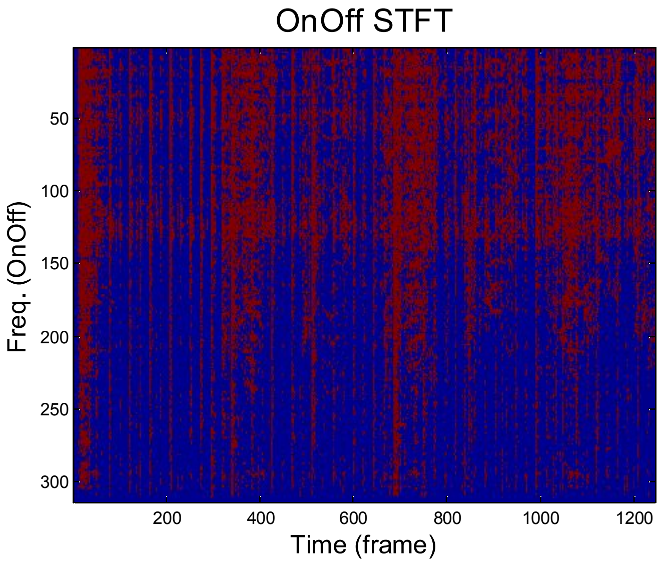


Figure 19. On-off filtered STFT spectrogram

Then, we eliminate harmonics of harmonious components using spectral median filtering. Note that temporal median filtering erased the abruptly appearing (and fast decaying) drum sounds. The spectral median filtering, however, regards the harmonics of the spectrum of the given frame as irregular ones and removes. It is clear that the peaky harmonics in the spectrum looks similar to the peaks of impulsive instruments [23]. The rough sound components, on the contrary, are apt to be continuous enough not to be eliminated by spectral median filtering. Furthermore, the less harmonious components from rough sounded instruments, such as electric guitars, can be also extracted with this process as a side effect. We welcome those accompanying components as well, because the roughness of partly harmonious instruments can be a good indicator about how arousing the song is. After summing those processed signal, we can get the feature which approximately shows the amount of rough sounds in the songs. From the Figure 20, we can find that the rough sound components are remained well while the harmonious components are removed, after on-off and spectral median filtering.

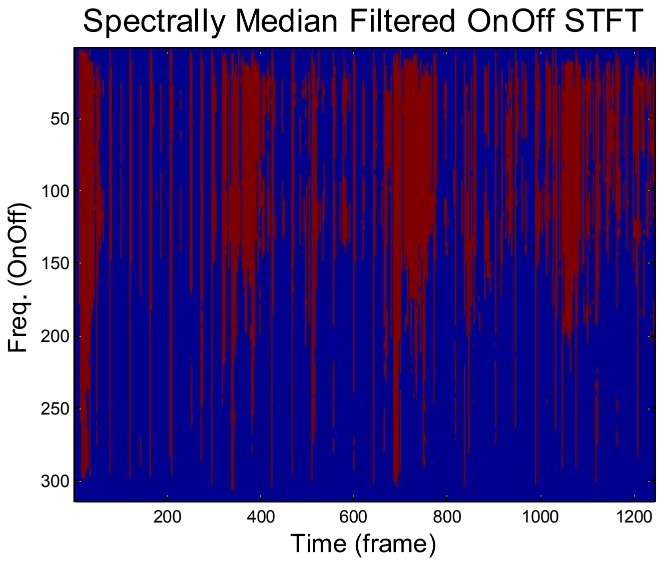


Figure 20. Spectral median filtering of on-off filtered STFT spectrogram

KAIST

Figure 21 shows the frame-by-frame summation results of the on-off and median filtered spectrogram. We propose these intensities as our feature for approximately representing the amounts and dynamics of rough sound components of the songs.

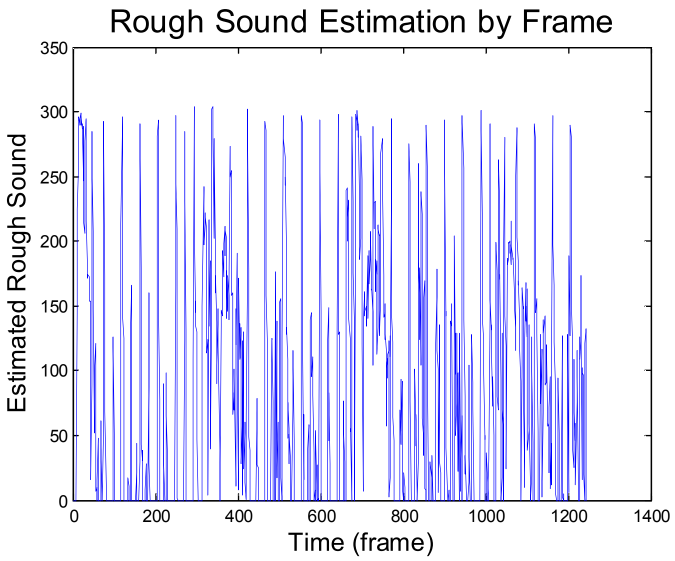


Figure 21. Frame-by-frame summation results of the on-off and median filtered STFT spectrogram

IV Experiments and Results

4.1 System Optimization

4.1.1 SVM Grid Search

The SVM is very popular and powerful, so that many music classification systems use it as their classifier. SVM finds the hyperplane with the support vectors, which consists of the samples nearest from the hyperplane. We call the distance between a support vector and the hyperplane the *margin*. SVM chooses the hyperplane which makes the margin maximized, because the larger margin lowers the generalization error of the classifier [24].

SVM can cope with both linearly separable data and linearly non-separable data, but it basically works like a linear classifier. However, real world data are not linearly separable in most cases, so Vapnick expanded SVM by adding the concept of error to the non-separable data [25]. Training errors and the margin have a trade-off relationship, so we need to choose the appropriate amount of error. SVM defines a variable, called C, which controls the size of error as a penalty. The performance of SVM depends pretty much on the proper value of C, so we need to find its optimal value. The larger the value of C, the lower the training error.

KAIST

SVM can be expanded to classify non-linear dataset by using the kernel trick [26]. It is widely known that the separation task can be easier in higher dimensions. However, the number of possible kernel is infinite, because anything can be a kernel if it satisfies basic property of kernel. Fortunately, there are popularly recommended kernel functions when we use SVM. The most highly recommended kernel function is

Radial Basis Function (RBF), so we use RBF kernel and linear one as well.

RBF kernel has following formula. We need to decide the best value of γ on the optimization process along with the cost C.

$$\exp(- \gamma *|x-x^T|^2) \tag{1}$$

At first, we investigate that the MARSYAS features and the proposed features are really necessary in AMC task since we do not know what can happen with the nonlinear kernels and raw signal in SVM. Maybe SVM can manage the raw signal and transform it into the high dimension, so that it can actually replace the feature extraction procedure. Our goal is to find the performance limitation of SVM without the help of feature extraction phase. We firstly use the STFT spectrogram only, with the temporal approximation technique in MARSYAS. Then, we consider the possiblity of performance improvement by putting feature vectors which extract some low-level and mid-level music features.

KAIST

To decide which values to choose for the penalty of error C and the γ of the kernel function, we used a grid search algorithm following the instructions in [27]. The grid search technique is a brute method which finds the best parameter set among every possible combination of parameters lying in the pre-defined ranges. We pick the parameters which yields the best classification accuracy using 3-fold cross validation on the training set. However, the deviation of the accuracy is too large according to the change of the folding points, so we shuffle 30 times per every parameters set to get the mean of them. This makes the optimal parameters more reliable.

In this thesis, we use *LIBSVM* package for the train-and-test part of our proposed system [28].

We used following 4 feature sets for the experiment.

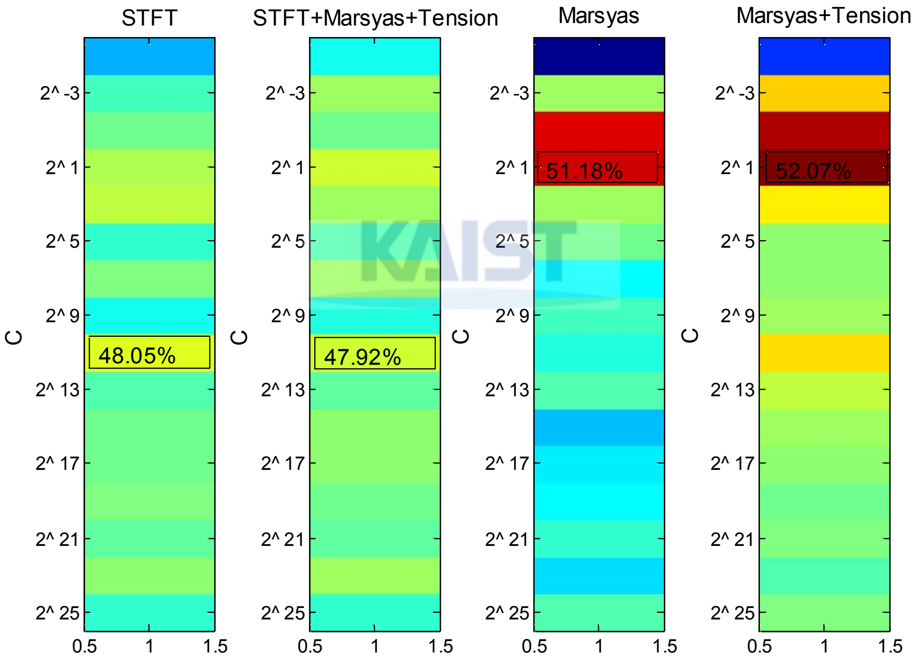
Feature Set 1: STFT

Feature Set 2: MARSYAS features

Feature Set 3: MARSYAS features and Chord tension

Feature Set 4: STFT, MARSYAS features, and Chord tension

MARSYAS features are the already included features in MARSYAS for its classification tasks: MFCC, spectral centroid, spectral flux and roll-off point.



KAIST

Figure 22. Optimization results of linear SVM with diverse values of C.

We can see that the optimization results of figure 22 say that the STFT only case does not reach the performance of the feature extracted cases. Compared with figure 23, note that the STFT case performs better in linear kernel than RBF since the dimension of STFT feature is too high to be affected by nonlinear kernel [27]. The MARSYAS features and chord tension features, however, does exceed the best classification accuracy of STFT case.

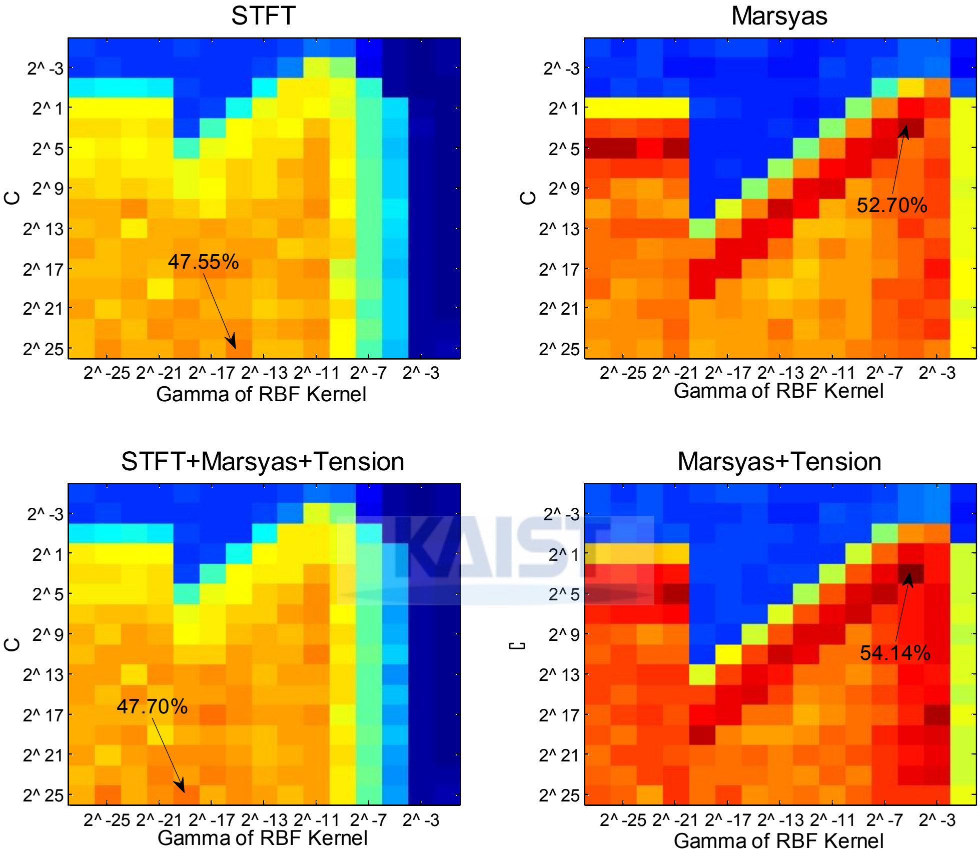


Figure 23. Optimization results of RBF SVM with diverse values of C and γ.

Table 4 summarizes the SVM optimization results. Using STFT spectrums only, we get the best result of 48.05% when the kernel is linear and C is 2^{11}. However, if we use the MARSYAS features, SVM resulted in 52.7% at best when the kernel is RBF, C is 2^3, and γ is 2^{-5}. Promisingly, we get the best result of all the experiments with the MARSYAS features plus our proposed chord tension feature, which is 54.14% with the same parameters with the MARSYAS only case. Note that adding the processed features to the raw STFT signal does not improve the results. We can conclude with this optimization experiments that we have to take more condensed features into consideration.

Features	**Kernel**	**C**	**γ**	**Accuracy**
STFT	Linear	2^{11}	-	48.05%
MARSYAS features	RBF	2^3	2^{-5}	52.70%
MARSYAS features and Chord tension	RBF	2^3	2^{-5}	**54.14%**
STFT, MARSYAS features, and Chord tension	Linear	2^{11}	-	47.93%

Table 4 Summarization of SVM optimization results

4.2 Evaluation Environment

KAIST

4.2.1 Data set

Three kinds of data sets are used with the experiments.

1) 132 exemplar songs which MIREX offers.
2) 185 balanced data set which consists of MIREX exemplar songs and additionally gathered songs for this thesis
3) 600 ground truth set which MIREX actually uses in the contest.

The first data set, which MIREX offers as an exemplar, consists of 132 famous pop songs. However, it is not balanced well: the numbers of songs in class 4 and 5 are relatively lower than the others. The confusion matrix of classification results shows that the songs in class 4 and 5 tend to be misclassified as class 1 or 2, which contains more number of songs. We, therefore, add some additional songs into each class to make them balanced. The second data set is the one with the first data set and the newly added songs. To match with the definition of the classes, we carefully chose the new songs from the already included musicians in each class. After testing with these exemplars, we finalize our proposed features, and then send our system to MIREX for evaluation with 600 ground truth data.

4.2.2 Evaluation Environment

1) MIREX evaluation environment

2) Our own evaluation environment

KAIST

In the MIREX evaluation environment, we concede that its 600 dataset and 3-fold cross validation do benefit the trustworthiness of contest. However, we want raise an issue about the folding point problem. It is clear that the classification accuracy depends on which point the folding is made. We cannot simply ignore the fluctuation caused by the different folding points because some systems might work better than the others only in the specific folding point and not in the others. MARSYAS system, for example, did not change its settings for the two MIREX AMC tasks in 2007 and 2008, but the classification accuracy changed from 58.20% to 61.50%. To overcome this problem, we devise our own evaluation environment with 3-fold cross validation which is done 100 times. During the 100 times of 3-fold cross validation tests, we also shuffle the training data to change the folding point randomly. Then, we take the mean of the 100 classification results, which is hopefully more robust than MIREX results considering the folding point problem. However, it is not possible to ignore the MIREX evaluation environment, since it really uses the verified massive data set in spite of its lack of consideration on folding point problem.

Note that our own evaluation environment is also used in MIREX evaluation environment with a nested structure, when it trains SVM classifiers. With a given training data, which is 2/3 of the 600 ground truth set per each folding, we check the performance of the SVM parameter sets by using iterative 3-fold cross validation again on them. Then, we choose the statistically best resulting configuration of SVM through 100 repetitions of our own 3-fold cross validation.

4.3 Evaluation Result

KAIST

	Predict 1	**Predict 2**	**Predict 3**	**Predict 4**	**Predict 5**
Real 1	61.97	17.76	6.00	10.78	3.49
Real 2	25.27	43.62	20.46	10.24	0.41
Real 3	12.14	12.03	61.92	9.73	4.19
Real 4	23.32	20.59	14.32	32.32	9.43
Real 5	25.73	5.35	8.08	15.81	45.03

(a) Marsyas Only (48.97% in average)

	Predict 1	**Predict 2**	**Predict 3**	**Predict 4**	**Predict 5**

Real 1	64.32	14.30	7.92	10.49	2.97
Real 2	23.65	46.57	19.00	9.76	1.03
Real 3	6.38	10.19	71.08	7.27	5.08
Real 4	22.73	20.57	15.16	32.57	8.97
Real 5	19.41	4.51	8.22	16.43	51.43

(b) Marsyas Only + Chord Tension (53.19% in average)

	Class1	**Class2**	**Class3**	**Class4**	**Class5**
Marsyas Only	62.97	43.62	61.92	32.32	45.03
+Chord Tension	64.32	46.57	71.08	32.57	51.43
Increment Rate	+1.35	+2.95	+9.16	+0.25	+6.4

(c) Increment rate per class

Table 5 Confusion matrices for Marsyas features and tension feature

KAIST

Table 5 shows the effectiveness of chord tension feature when it is added to the MARSYAS features. We use the second data set, which is balanced well, for this and two following experiments. With this small set of exemplar songs, the net performance increased from 48.97% to 53.19%. The confusion matrix represents that the classification rate of class 3 enhanced by around 9%. It also says that each result of the other class also increased. Figure 24 gives us another explanation of this experiment that the means of the texture windowed means of tension feature, comparatively. We can check that the class 3 songs marked lower mean-mean tension values compare to the others, which are mainly made of sentimental songs.

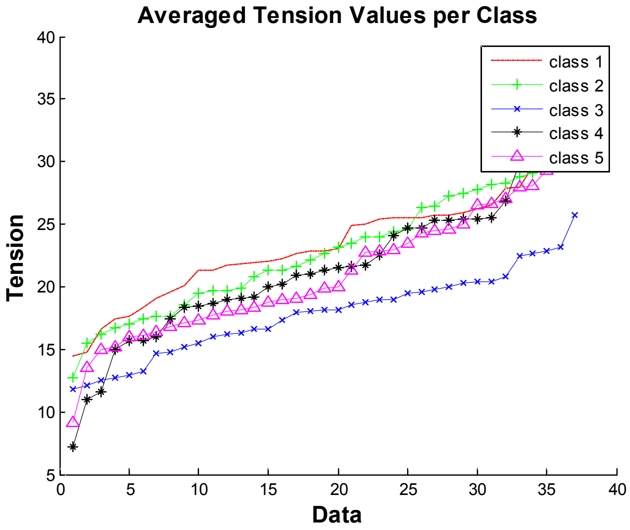


Figure 24. Distribution of averaged tension values per class

Table 6 shows the performance growth by adding the rough sound feature to the MARSYAS features. It increases the net classification accuracy from 48.97% to 50.41%. Figure 25 depicts more about this feature. The means of standard deviations of texture windowed songs clearly distinguishes class 5 from the others. We can explain this results that the rough sound, such as drums, are played more dynamically in the songs of class 5, where the aggressive and fiery songs are included.

KAIST

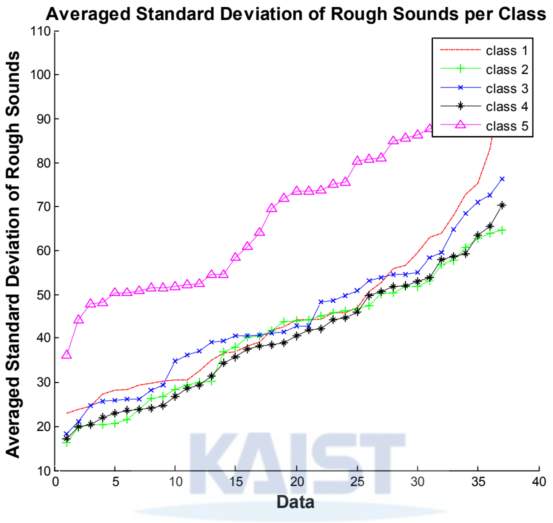
	Predict 1	**Predict 2**	**Predict 3**	**Predict 4**	**Predict 5**
Real 1	63.05	16.92	5.24	9.97	4.81
Real 2	24.24	44.70	21.51	9.03	0.51
Real 3	11.14	13.00	64.03	8.43	3.41
Real 4	21.65	21.81	14.08	33.65	8.81
Real 5	25.38	5.38	8.76	13.89	46.59

(a) Marsyas Only + Rough Sound (50.41% in average)

	Class1	**Class2**	**Class3**	**Class4**	**Class5**
Marsyas Only	62.97	43.62	61.92	32.32	45.03
+ Rough Sound	63.05	44.70	64.03	33.65	46.59
Increment Rate	+0.08	+1.08	+2.11	+1.33	+1.56

(b) Increment Rate

Table 6 Confusion matrix of Marsyas features and rough sound Feature



KAIST

Figure 25. Distribution of avearged standard deviations of rouph sounds per class

Table 7 describes the performance of both the tension and rough sound features. The performance rises from 48.97% to 54.69% when the two proposed features are included to the MARSYAS features. The confusion matrix also shows that the discrimination ability is enhanced especially on class 3 and class 5 data.

	Predict 1	Predict 2	Predict 3	Predict 4	Predict 5
Real 1	65.92	12.92	8.19	8.78	4.19
Real 2	21.92	48.41	19.24	8.73	1.70
Real 3	6.73	10.16	72.08	6.30	4.73
Real 4	19.73	22.08	13.08	35.73	9.38
Real 5	19.92	5.59	8.00	15.16	51.32

(a) Marsyas Only + Chord Tension + Rough Sound (54.69% in average)

	Class1	Class2	Class3	Class4	Class5
Marsyas Only	62.97	43.62	61.92	32.32	45.03
+Chord Tension	65.92	48.41	72.08	35.73	51.32
Increment Rate	+2.95	+4.79	+10.16	+3.41	+6.29

(b) Increment Rate

Table 7 Confusion matrix of Marsyas features and chord tension and rough sound Feature

KAIST

Finally, the experiment with the 600 ground truth data of MIREX is done. Table 8, 9, and 10 show the performance of each fold, net performance and confusion matrix of four feature sets, respectively. The four names of system KHC1 to KHC4 mean MARSYAS features, MARSYAS features plus tension feature, MARSYAS features plus rough sound features, and all of them, respectively. In the first fold of MIREX evaluation environment, our systems perform over 70% in average. For all the 3-fold cross validation, they exceed over 60% in average. We can see that the newly proposed features of this thesis improved the classification performance when they are added to the MARSYAS features. Furthermore, our systems are superior to the systems which are submitted in 2008 and 2007's MIREX AMC contests.

Classification Fold	KHC1	KHC2	KHC3	KHC4
1	69.5%	73.5%	71.0%	74.0%
2	62.0%	61.0%	64.0%	59.5%
3	58.5%	57.5%	56.0%	59.0%

Table 8. Performance result of each fold with the 600 ground truth data of MIREX

Participant System	Mean Accuracy
KHC1	63.33%
KHC2	64.00%
KHC3	63.67%
KHC4	64.17%

KAIST

Table 9. Mean accuracy with the 600 ground truth data of MIREX

Class	KHC1	KHC2	KHC3	KHC4
1	50.00%	52.50%	48.33%	51.67%
2	55.00%	54.17%	53.33%	53.33%
3	83.33%	82.50%	82.50%	80.83%
4	52.50%	56.67%	55.00%	56.67%
5	75.83%	74.17%	79.17%	78.33%

Table 10 Confusion matrix with the 600 ground truth data of MIREX

Table 11 shows that our proposed systems with chord tension and rough sound features outperform the best one in the recent two year's MIREX AMC task.

AMC System	**Accuracy(%)**
KHC4 2009	**64.17**
KHC2 2009	**64.00**
KHC3 2009	**63.67**
GP 2008	63.67
GT 2007	61.50
CL 2007	60.50
TL 2007	59.67
GT 2008	58.20
ME 2007	57.83
CL 2008	56.00
ME 2008	50.33

KAIST

Table 11. Comparison of proposed systems with the top-ranking recent two years' MIREX submissions

V Conclusions

This thesis proposed novel mid-level music features and verified their performance with AMC tasks. Based on music theory and appropriate filtering, we devised a method for directly extract chord tension from the signals. We also proposed a compact rough sound extraction method for taking the amount of inharmonious components into account. We examined the intuition about the proposed features empirically, and then concluded that they help classify some classes better. In order to convince the performance of the proposed AMC system we double checked it with the larger dataset, which is actually used for MIREX AMC contest. With the fair experiment by the MIREX committee, we verified that the proposed system marked the first among the all submitted AMC systems in recent two years.

The proposed mid-level features captured musical properties quite well, yet they need improvement. The simplicity of rough sound extraction is clearly a virtue, but we expect more accurate classification rate if we adapt the sophisticated source separation technique. We have concluded the other music-related frequency representation, like PCP, is inappropriate, but, more refined one could help the chord tension extraction a lot. Furthermore, the other mid or high-level features are possible to improve the performance if it is well-defined and able to be extracted elaborately. Finally, we guess that some proper dimension reduction techniques, such as Principle Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Locality Preserving Projections (LPP), can be used for condensing the features more and enhancing the performance.

KAIST

崔 嘉 晛

상위 레벨 음악 특성을 사용한 음악 감정 분류 성능 향상

음악을 검색하는 일반적인 방식은, 곡 제목, 가사, 음악가와 같은 텍스트 형식의 정보이다. 그러나, 디지털 음악 기술이 발전하고 음악 데이터베이스 용량이 커짐에 따라, 기존의 음악 검색방식의 한계점이 드러나게 되었다. 곡명이나 가사는 기억나지 않고 멜로디만 생각나는 경우나, 특정 상황에 맞는 음악 리스트가 필요한 경우에는 보다 발전된 음악 검색 기술의 도움이 필요하다. 이런 새로운 요구를 반영한 음악 검색 방식의 예로 허밍을 통한 음악 검색, 유사 음악 검색, 장르나 감정에 기반한 음악 검색을 들 수 있다. 이 중 대량의 음악에서 음악을 분류하는 시스템의 경우, 대량의 음악에 일일이 장르나 감정 태그를 붙이는 것은 거의 불가능하다. 따라서, 자동으로 음악을 분류하는 기술의 필요성이 대두된다. 본 연구에서는 감정에 맞게 음악을 자동으로 분류하는 시스템의 성능 향상을 이룰 수 있는 방법에 대해 고찰한다.

KAIST

우선 음악 감정 분류 시스템에 대해 연구할 때, 공학적으로 다루기에는 다소 모호한 개념인 "감정"을 형식화할 필요가 있다. 그리고, 특정 음악이 어떤 감정을 유발하는지에 관해, 형식화된 감정 값과 음악 콘텐츠 간의 신뢰할 수 있는 매핑이 필요하다. 본 연구에서는 MIREX 프레임워크가 제공하는 5 가지 감정군과 그 감정군에 맞게 태그를 붙인 600 곡의 DB 를 사용하였다. MIREX 에서 대규모의 감정 카테고리로부터 군집화한 감정군과 다수의 사람들이 참여하고 그 중 2/3 이상의 동의를 얻은 음악으로만 구축된 600 곡에 대한 음악 감정 관계는 연구의 신뢰성을 높여준다. 또한, 신뢰성 있는 기존 연구인 MARSYAS 를 참고 시스템으로 활용하여, MARSYAS 가 가지고 있는 우수한 성능과 높은 재현성을 본 연구에 반영하였다.

또한 본 연구에서는, 음악적인 요소를 좀 더 많이 반영하는 새로운 특징 개발의 필요성을 타진하기 위해, 참고 시스템에서 사용한 잘 알려진 특징들과, 특징 추출 과정이 생략된 주파수 영역 데이터를 비교하였으며, 이를 위해 SVM 최적화 과정을 진행하였다. 그 결과, 보다 높은 수준의 음악 특성을 추출하는 특징 벡터 개발의 필요성을 발견하였다.

이러한 검증 결과를 바탕으로, 본 연구에서는 두 개의 중간 수준의 음악 특성을 추출하는 특징 벡터를 개발하였다. 새로 제안한 특징 벡터는 코드의 긴장도와 거친 소리를 뽑아내는 것이다. 코드 긴장도의 경우 감정의 두 축 중 긴장도에 영향을 미치는 요인이다. 본 연구에서는 에러율이 높은 코드 인식 기술이나 자동 채보 기술을 사용하지 않고 주파수 분석 정보로부터 바로 코드의 긴장도를 추출하는 방법을 고안하였다. 다음 특징 벡터는 거친 소리 부분을 추출하는 피쳐로, 거친 소리란 드럼이나 왜곡된 전자기타처럼 음악에서 노이즈 성분이 강한 부분을 말한다. 본 연구에서는 기존의 음악 음원 분리 기술들에 비하여 계산 복잡도 측면에서 경쟁력있고 잘 동작하는 거친 소리 추출 방법을 제안하였다.

새로운 특징 벡터의 도입을 통해 개선한 음악 감정 분류 시스템은 MIREX 에서 제공하는 음악 감정 정답 데이터 베이스로 평가하였다. 평가 결과 본 논문에서 제안한 두 개의 중간 수준 음악 특성을 뽑아내는 특징 벡터를 사용한 음악 감정 분류 시스템은 최근 2 년간 MIREX 에 출품한 모든 시스템의 성능을 능가하는 결과를 보여주었다.

KAIST

References

[1] J. Downie, D. Byrd and T. Crawford, “Ten Years of ISMIR: Reflections on Challenges and Opportunities”, In Proceeding of International Society for Music Information Retrieval Conference (ISMIR), Kobe, Japan, October 26-30, 2009.

[2] C. Laurier, M. Sordo and P.Herrera, “Mood Cloud 2.0: Music mood browsing based on social networks”, In Proceeding of International Society for Music Information Retrieval Conference (ISMIR), Kobe, Japan, October 26-30, 2009.

[3] O. Celma and P. Lamere, “Music Recommendation Tutorial”, In Proceeding of International Society for Music Information Retrieval Conference (ISMIR), Vienna, Austria, September 23-27, 2007.

[4] R. Thayer, “The Biopsychology of Mood and Arousal”, New York: Oxford University Press, 1989.

[5] Audio Chord Detection Results in 2007’s MIREX: http://www.music-ir.org/mirex/2008/index.php/Audio_Chord_Detection_Results

[6] M. Yoo and I. Lee. “Musical Tension Curves and its Applications”, In Proceeding of International Computer Music Conference (ICMC), New Orleans, U.S.A., November 6-11, 2006,

[7] MIREX: http://www.music-ir.org/mirexwiki/index.php/Main_Page

[8] X. Hu and J. Downie, “Exploring mood metadata: Relationships with genre, artist and usage metadata,” In Proceeding of International Society for Music Information Retrieval Conference (ISMIR), Vienna, Austria, September 23-27, 2007.

[9] X. Hu, J. Downie, C. Laurier, M. Bay and A. Ehmann, “The 2007 Mirex audio mood classification task: lessons learned”, In Proceeding of International Society for Music Information Retrieval Conference (ISMIR), Philadelphia, Pennsylvania, U.S.A., September 14-18, 2008.

KAIST

[10] G. Tzanetakis, “Marsyas submissions to MIREX 2007”, MIREX, 2007.

[11] G. Tzanetakis, “Musical Genre Classification of Audio Signals”, IEEE Transactions on Speech and Audio Processing, Vol. 10, No. 5, pp. 293-302, July, 2002.

[12] S. Davis and P. Mermelstein, “Experiments in syllable-based recognition of continuous speech,” IEEE Transactions on Acoustics, Speech, Signal Processing, vol. 28, pp. 357–366, Aug. 1980.

[13] T. Fujishima, “Realtime chord recognition of musical sound: a system using common lisp music”, In Proceeding of International Computer Music Conference (ICMC), pages 464–467, Bejing, China, 1999.

[14] G. Wakefield, “Mathematical representation of joint time chroma distributions”, In Proceeding of SPIE conference on Advanced Signal Processing Algorithms, Architecture and Implementations, pages 637–645, Denver, Colorado, USA, 1999.

[15] G. Peeters, “A Generic Training and Classification System for MIREX 08”, MIREX, 2008

[16] H. Papadopoulos and G. Peeters, “Large-scale study of chord estimation algorithms based on chroma representation and hmm,” in Proceeding of International Workshop on Content-Based Multimedia Indexing, Bordeaux, France, 2007.

[17] A. Sheh and D. Ellis, “Chord segmentation and recognition using EM-trained hidden Markov models,” in Proceeding of International Society for Music Information Retrieval Conference (ISMIR), pp. 185–191, 2003.

[18] J. Bello and J. Pickens, “A robust mid-level representation for harmonic content in music signals,” in Proceeding of International Society for Music Information Retrieval Conference (ISMIR), pp. 304–311, 2005.

[19] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the CUIDADO project", CUIDADO I.S.T. Project Report,

KAIST

2004.

[20] H.-T. Cheng et al., “Automatic chord recognition for music classification and retrieval”, In Proceeding of IEEE International Conference on Multimedia and Expo., pp. 1505-1508, 2008

[21] J. Brown, "Calculation of a constant Q spectral transform", Journal of the Acoustical Society of America, Vol. 89, No. 1, 425-434

[22] M. Helén, T. Virtanen, “Separation of Drums From Polyphonic Music Using Non-Negative Matrix Factorization and Support Vector Machine”, in Proceeding of 13th European Signal Processing Conference, Antalaya, Turkey, 2005.

[23] E. Tsunoo, N. Ono and S. Sagayama, “Musical Bass-Line Pattern Clustering and Its Application to Audio Genre classification”, in Proceeding of International Society for Music Information Retrieval Conference (ISMIR), Kobe, Japan, October 26-30, 2009.

[24] T. Hastie, R. Tibshirani and J. Friedman, “The elements of statistical learning: data mining, inference, and prediction”, 2nd Edition, Springer, 2009

[25] V. Vapnik, “The Nature of Statistical Learning Theory”, Springer-Verlag, 1995

[26] B. Boser, I. Guyon and V. Vapnik, “A training algorithm for optimal margin classifiers”, In D. Haussler, editor, 5th Annual ACM Workshop on COLT, pages 144-152, Pittsburgh, PA, 1992. ACM Press

[27] C.-W. Hsu, C.-C. Chang, C.-J. Lin, “A practical guide to support vector classification”, http://www.csie.ntu.edu.tw/cjlin/papers/guide/guide.pdf

[28] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," 2001, Software available at: http://www.csie.ntu.edu.tw/~cjlin/libsvm

KAIST