MusCat: A Music Browser Featuring Abstract Pictures and Zooming User Interface

1st Author	2nd Author	3rd Author
1st author's affiliation	2nd author's affiliation	3rd author's affiliation
1st line of address	1st line of address	1st line of address
2nd line of address	2nd line of address	2nd line of address
Telephone number, incl. country code	Telephone number, incl. country code	Telephone number, incl. country code
1st author's email address	2nd E-mail	3rd E-mail

ABSTRACT

Today many people store music media files in personal computers or portable audio players, thanks to recent evolution of multimedia technologies. The more music media files these devices store, the messier it is to search for tunes that users want to listen to. We propose MusCat, a music browser to interactively search for the tunes according to features, not according to metadata (e.g. title, artist name). The technique firstly calculates features of tunes, and then hierarchically clusters the tunes according to the features. It then automatically generates abstract pictures, so that users can recognize characteristics of tunes more instantly and intuitionally. It finally visualizes the tunes by using abstract pictures. The technique enables intuitive music selection with the zooming user interface.

1. INTRODUCTION

Recently many people listen to the music by using personal computers or portable players. Numbers of tunes stored in our computers or players quickly increase due to the increase of sizes of memory devices or hard disk drives. User interfaces therefore become more important for users to easily select tunes users want to listen to.

Here we think that the procedure to search for the tunes would be more enjoyable, if we develop a technique to interactively search for tunes, not based on metadata but based on features. We usually select tunes based on their metadata, such as titles, artist names, and album names. On the other hand, we may want to select tunes based on musical characteristics depending on situations. For example, we may want to listen to graceful tunes at quiet places, loud tunes at noisy places, danceable tunes at enjoyable places, and mellow tunes at night. Or, we may want to select tunes based on feelings. For example, we may want to select “something speedy/slow” or “something major/minor” tunes based on feelings. We think features are often more informative rather than metadata, to select tunes based on situations or feelings. However, it is not very intuitive if we show feature values of tunes just as numeric characters. We think illustration of features may help intuitive tune selection.

This paper presents MusCat, a music browser featuring abstract pictures and zooming user interface. It visualizes collections of tunes by abstract pictures, based on features, not based on metadata. The technique presented in this paper firstly calculates features of tunes, and hierarchically clusters the tunes according to the features. It then automatically generates abstract pictures for each tune and cluster, so that users can recognize characteristics of tunes more instantly and intuitionally. It finally displays the tunes and their clusters by using the abstract pictures.

We apply an image browser CAT [1] to display a set of abstract pictures. CAT supposes that hierarchically clustered pictures are given and representative pictures are selected for each cluster. CAT places the set of pictures onto a display space by applying a rectangle packing algorithm that maximizes the display space utilization. CAT provides a zooming graphical user interface, which displays representative pictures of all clusters while zooming out, and pictures in the specific clusters while zooming in, to effectively browse the pictures. We call the user interface “MusCat”, as an abbreviation of “Music CAT”, which enables intuitive music selection with its zooming user interface.

2. RELATED WORK

2.1 Music and Sensitivity

There have been several works on expression of musical features as adjectives. For example, Yamawaki et al. [2] presented that people recognize three pairs of adjectives: heavy - light, speedy - slowly, and powerful - unpowerful, as impression of music, in a part of their study of the correspondence analysis. Similarly, our technique also assigns two pairs of adjectives to musical features.

2.2 Color and Sensitivity

There have been several studies on the relationship between colors and sensitivity, and our work is on the top of such studies. Color system [3] expresses impressions of colors by arranging them onto a two dimensional sensitivity space, which has warm-cool and soft-hard axes. The color system supposes to place homochromous and polychromous colors onto the sensitivity space: it arranges homochromous colors onto a limited region of the sensitivity space, while it arranges polychromous colors covering whole the sensitivity space. Therefore, we think that

SAC’11, March 21-25, 2011, TaiChung, Taiwan.

impression of music can be adequately expressed by using polychrome colors rather than by homochromous colors.

2.3 Combination of Tunes and Pictures

There have been several techniques on coupling tunes and pictures. MIST [4] is a technique to assign icon pictures to tunes. As a preparation, MIST requires users to answer the questions about conformity between sensitive words and features of sample tunes or icons. MIST then learns correlations among the sample tunes, and couples the tunes and icons based on the learning results.

Kolhoff proposed Music Icons [5], a technique to select abstract pictures suited the tunes based on their features. The technique presented in this paper is also based on the features of tunes, but it focuses on the automatic generation of abstract pictures.

2.4 Feature-based Music Analysis and Retrieval

There have been several techniques for music analysis and retrieval. For example, Lie et al. [6] presented a hierarchical framework that automates the task of mood detection from acoustic music data, by following some music psychological theories in western cultures. It extracts three feature sets, intensity, timbre, and rhythm, to represent the characteristics of music clips.

2.5 User Interface for Music

User interface is very important for interactive music retrieval, and therefore many techniques have been presented. Goto et al. presented Musicream [7], which enables enjoyable operations to group and select tunes. Lamere et al. [8] presented "Search inside the Music," which applies a music similarity model and a 3D visualization technique to provide new tools for exploring and interacting with a music collection.

2.6 Image Browser

Image browser is an important research topic, because numbers of pictures stored in personal computers or image search engines are drastically increasing. CAT (Clustered Album Thumbnail) [1] is a typical image browser that supports an effective zooming user interface. CAT supposes that hierarchically clusters pictures are given and representative pictures are selected for each cluster. CAT firstly packs thumbnails of the given pictures in each cluster, and encloses them thumbnail by rectangular frames to represent the clusters. CAT then packs the rectangular clusters and encloses them by larger rectangular frames. Recursively repeating the process from the lowest to the highest level of hierarchy, CAT represents the hierarchically clustered pictures.

In addition, CAT has a zooming user interface, as shown in Figure 1. CAT displays representative images instead of rectangular frames while zooming out, as the initial configuration. On the other hand, CAT displays image thumbnails while zooming in the specific clusters. CAT enables to intuitively search for interesting images, by briefly looking at the all representative images, then zooming into the clusters displayed as interested representative images, and finally looking at the thumbnail images in the clusters.

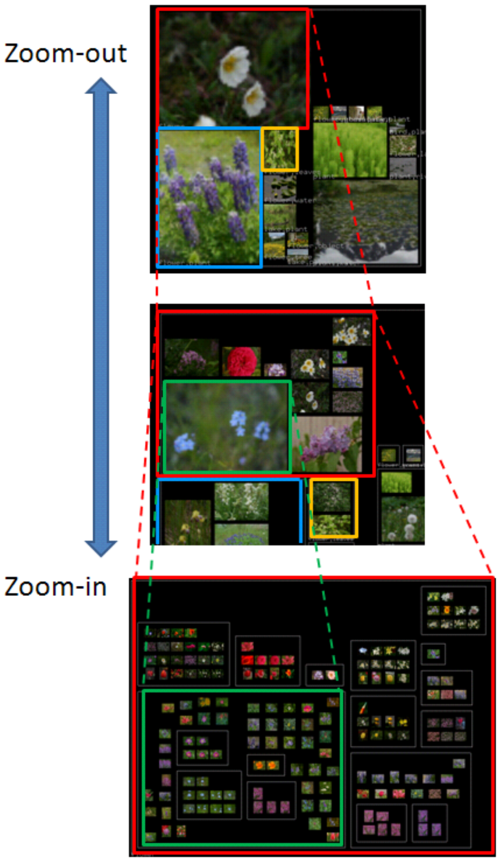

Figure 1. Zooming operation of an image browser CAT.

3. PRESENTED TECHNIQUE

Our technique consists of the following four steps: (1) calculation of features from music media files, (2) clustering of tunes based on the features, (3) generation of abstract pictures, and (4) visualization of tunes by using abstract pictures.

Our current implementation uses acoustic features (MFCC: Mel-Frequency Cepstrum Coefficient) for clustering, because we assumed that acoustic-based division is intuitive to briefly select tunes based on situation, such as "quiet tunes for quiet spaces", or "loud tunes for noisy spaces." On the other hand, it uses musical features for abstract image generation, because we assume that more information may be needed to select specific tunes from clusters of similarly sounding tunes.

3.1 Music Feature Extraction

There have been various techniques to extract music features, and some of them have been components of commercial products or open source software. Our current implementation uses features calculated by Marsyas [9] and MIRtoolbox [10]. It uses means and standard deviations of nine bands of MFCC calculated by Marsyas, for clustering and abstract image generation for clusters. We preferred to use Marsyas for MFCC calculation just because it had more variables as results of MFCC calculation. Also, our implementation uses normalized features shown in Table 1 calculated by MIRtoolbox, for abstract image generation for tunes.

It may be sometimes difficult to express musical characteristics by single feature value, because features may change gradually or suddenly during a tune. Our current implementation calculates features from randomly selected 15 seconds of the tunes. In the future, we would like to extend our implementation so that we can select the most characteristic features from whole parts of the tunes.

Table 1. Music features we apply in our experiments.

Feature	Explanation
RMS energy	Root-mean-square energy which represents a volume of the tune.
Tempo	Tempo (in beats per minute).
Roll off	Frequency which takes 85% of total energy, by calculating the sum of energy of lower frequencies.
Brightness	Percentage of energy of 1500Hz or higher frequency.
Spectral irregularity	Variation of tones.
Roughness	Percentage of energy of disharmonic frequency.
Mode	Difference of energy between major and minor chords.

3.2 Clustering

Next, the technique hierarchically clusters music files based on means and standard deviations of MFCC values. There are many clustering techniques for multi-dimensional datasets (e.g., nearest neighbor, furthest neighbor, group average, centroid, median, Ward.) We experimentally applied various clustering techniques for our own collection of tunes, and compared the results by carefully looking at the dendrograms. As a result of our observation, we selected Ward method as a clustering algorithm, because it successfully divides a set of tunes into evenly sized clusters. Figure 2 shows the comparison of dendrogram among three clustering algorithms.

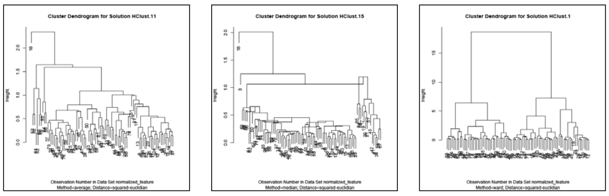

Figure 2. Comparison of dendrogram. (Left) group average method. (Center) median method. (Right) Ward method.

3.3 Abstract Picture Generation for Tunes

Our technique generates abstract pictures to express tunes, and displays so that users can intuitively select the tunes. It generates the abstract pictures for each tune, and for each cluster. It calculates the average of feature values for each cluster in order, to generate the abstract pictures of the clusters.

We believe the approach is effective as discussed below. First reason is that visual and music words are often related. Actually, many music works depict scenery that artists looked and imagined. Also, some painting artists expressed emotion while listening to the music as abstract pictures [8]. Second reason is that humans have synesthesia [11]. Impression of colors is related to impressions of sound, because some people have perception so called "the colored hearing," which associates colors by listening to the music. Third reason is that impressions of sounds and colors are often expressed by same adjectives. Therefore, we think that music can express through the abstract pictures considering colors to visualize the music.

This section describes our implementation to automatically generate abstract pictures. Currently it is just designed based on our subjective, but we do not limit the abstract pictures to the following design.

3.3.1 *Example of Color Assignment*

Our technique firstly assigns colors to the objects in abstract pictures. As mentioned previously, this technique uses polychrome colors because the polychrome color arrangement can express impression richer than the monochrome color arrangement. The technique selects colors of abstract pictures based on color image scale [3]. It is a color system that distributes combination of three colors in a two dimensional space, so called "sensibility space", which has the warm-cool and the soft-hard axes, as shown in Figure 3. The technique distributes tunes into this sensibility space, and assigns the colors corresponding in the sensitivity space to the tunes.

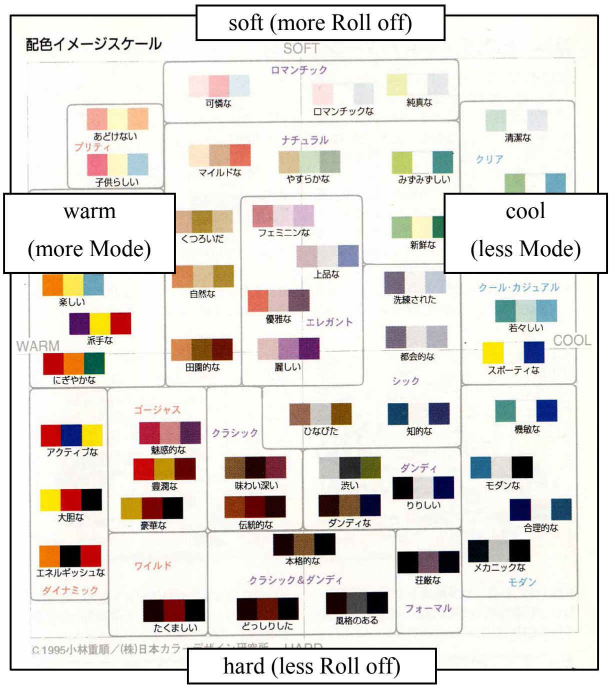

Figure 3. Sensitivity space based on color image scale.

Here, let us discuss which features match to the warm-cool and the soft-hard axes. We feel major chords express positive impression similar to bright and warm colors, while on the other hand, minor chords express negative impression similar to dark and cold colors. Based on the feeling, we assign Mode is to warm-cool axis. Similarly, we assign Roll off to soft-hard axis. We think listeners often use substitute adjective words such as "light" or "soft" for the impression of music, and often these impression is related to frequency-based tone balances.

Our current implementation places many sample colors onto the sensitivity space. Calculating Mode and Roll off of a tune, our implementation places the tune onto the sensitivity space, and selects the color closest to the tune. Here, let the position of the tune as (m_{wc}, m_{sh}), and the position of a color set as (c_{wc}, c_{sh}). Our implementation selects the color set that satisfies the following formula:

$$\min(\sqrt{(m_{wc} - c_{wc})^2 + (m_{sh} - c_{sh})^2})$$

3.3.2 Example of Abstract Picture Design

This section describes an example of design of abstract pictures based on music features. Our design first generates the following three layers, 1) gradation layer, 2) a set of circles, and 3) a set of stars, as shown in Figure 4.

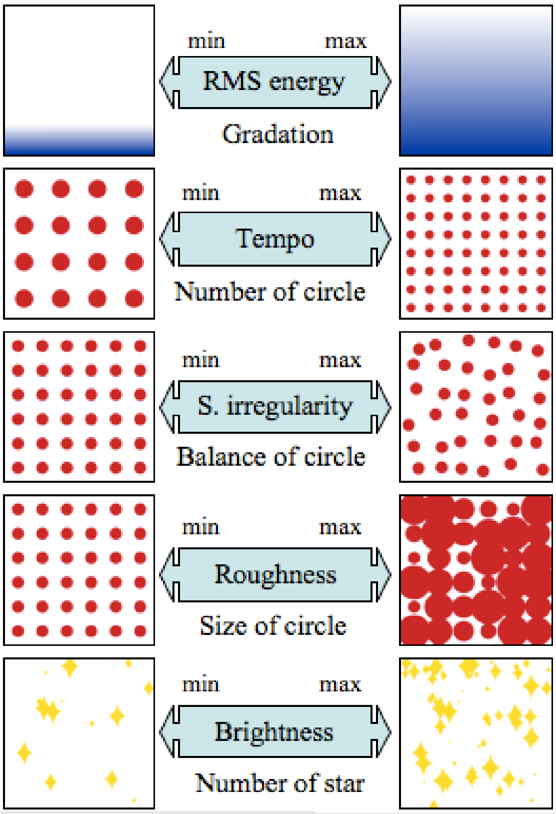

Figure 4. Automatic generation of three layers of images based on music features.

We assign RMS energy to the gradation layer. We would like to represent power, weight, and broadening by the gradation, and think that RMS energy is the most suitable feature for this representation.

We assign Tempo, Spectral irregularity, and Roughness, to the generation of orthogonally arranged circles. We would like to represent frequency of rhythm by the number of circles, irregularity and variation of music by irregularity of circles. We think Tempo, Spectral irregularity, and Roughness are the most suitable features for this representation.

We assign Brightness to the number of randomly placed stars. We expect many of users will associate bright music from the stars, and therefore we think Brightness is the most suitable feature for this representation.

After generating the three layers, the technique finally synthesizes the three layers to complete the abstract picture generation, as shown in Figure 5.

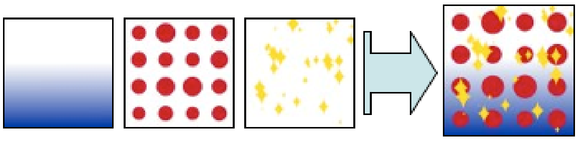

Figure 5. Abstraction picture synthesis from three layers of images.

3.4 Abstract Picture Generation for Clusters

Our technique generates another design of abstract pictures for clusters. It simply represents mean values of nine bands of MFCC as colored squares. Figure 6 shows an illustration how our technique generates abstract pictures. Our implementation defines nine colors for the bands based on the soft-hard axes shown in Figure 3. It assigns softer colors to higher bands, and harder colors to lower bands. It calculates the average values of the mean values of the tunes for each cluster, and calculates the sizes of the colored squares as proportional to the average values. The pictures denote acoustic textures of tunes in the clusters.

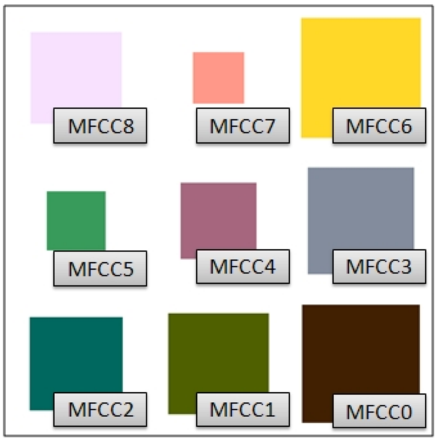

Figure 6. Illustration of abstract picture generation for clusters.

3.5 Image Browser CAT as a Music Browser

The technique displays a set of abstract picture by applying the image browser CAT [1]. We extend CAT so that we can use CAT as a music browser, where we call the extended CAT as "MusCat," as an abbreviation of "Music CAT".

Figure 7 is an example snapshot of MusCat. Initially MusCat shows all abstract pictures of clusters while zooming out, as shown in Figure 7(Lower). When a user prefers a picture and zooms in it, MusCat turns the abstract pictures for clusters to abstract pictures for tunes, as shown in Figure 7(Upper). Users can select a tune and play it by double-clicking the corresponding picture.

Figure 7 shows characteristic clusters (a), (b), and (c). The abstract image of cluster (a) denotes that the lowest frequency band (MFCC 0) of its tunes is respectively large. The abstract images of the three tunes in the cluster (a) have size-varying and un-aligned circles, and respectively more stars. These images denote that the tunes in cluster (a) have loud low and high frequency sounds, and respectively more disharmonic sounds. Actually, two of the three tunes are dance music that have loud Bass Drum beats, and backing of electric disharmonic sounds.

The abstract image of cluster (b) denotes that low and high frequency bands (MFCC 0, 1, 2, 7, and 8) are extremely small. The abstract images of the two tunes in the cluster (b) have aligned circles, and less number of starts. Colors of circles are different between the two images. Actually, the two tunes are Japanese traditional folk music played by old wood wind instruments, without bass part of high-tone percussions. One of the tunes plays major scale, and the other plays minor scale. That is why colors of two pictures of the tunes are much different.

The abstract image of cluster (c) denotes that specific two frequency bands (MFCC 1 and 3) are large, but others are quite small. The abstract images of the two tunes in the cluster (c) have smaller number of well-aligned, equally-sized circles. Actually, the two tunes are slow female vocal songs, and these backings are only piano or Japanese traditional strings. That is why the abstract image of the cluster denotes that specific two frequency bands are large, and the abstract images of the tunes have less number of circles.

As above mentioned, users of MusCat can select clusters of tunes based on acoustic features by specifying the abstract images of clusters, and then select tunes based on musical features. We think this order is reasonable: users can firstly narrow down the tunes based on their situations: for example, quiet tunes for quiet places, loud tunes for noisy places, and so on. They can select features of tunes in the specific clusters based on their feelings.

Though our original concept of MusCat supposes to show different abstract images for clusters and tunes, it is also possible to generate abstract images of tunes based on MFCC, as well as those of clusters. Figure 8 shows an example of abstract images of tunes generated based on MFCC.

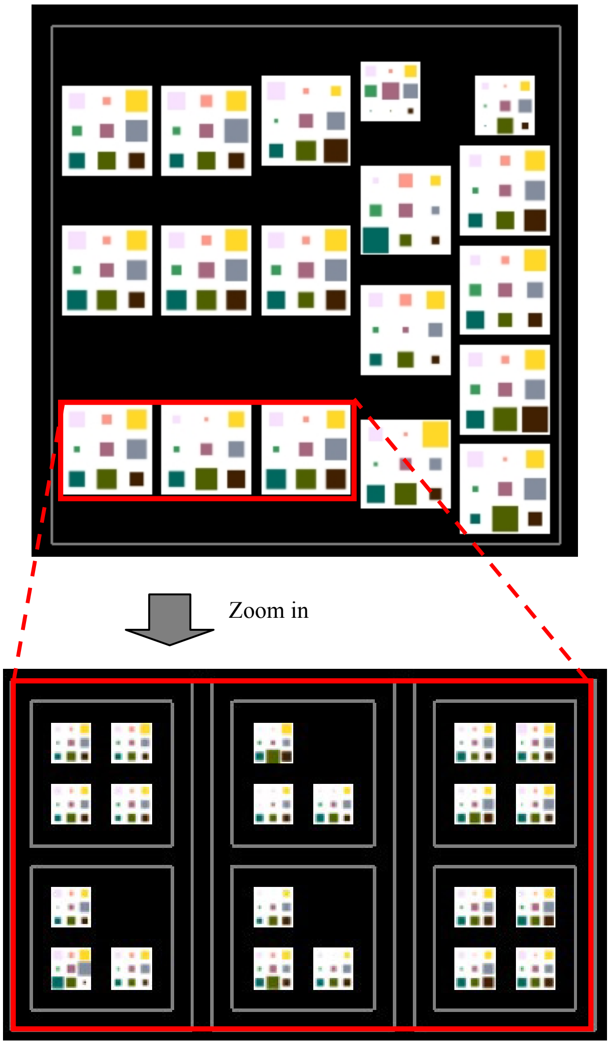

Figure 8. Example of visualization by MusCat. Abstract images of tunes are generated similar to those of clusters.

4. EXPERIMENTS

This section describes our experiments using the presented technique. We used Marsyas [9] and MIRtoolbox [10] for feature extraction, and R package for clustering. We implemented abstract picture generation module in C++ and executed with GNU gcc 3.4. We implemented MusCat in Java SE and executed with JRE 1.6. We applied 88 tunes which are categorized into 11 genres (pops, rock, dance, jazz, latin, classic, march, world, vocal music, Japanese, and a cappella), provided by RWC Music Database [12].

We had a user evaluation with 15 examinees to examine the validity of abstract pictures. We also asked the examinees to play with MusCat and give us comments or suggestions.

4.1 Suitability of Abstract Picture of Tunes

We showed 12 tunes and their abstract pictures to the examinees. We then asked to evaluate the suitability of the pictures for the tunes by 5-point scores, where "5" denotes suitable, and "1" denotes unsuitable. Table 2 denotes the statistics of the evaluation.

Table 2. Evaluation of abstract pictures.

	1 (suitable)	2	3	4	5 (unsuitable)
Tune 1	1	11	2	1	0
Tune 2	1	3	7	4	0
Tune 3	0	0	4	7	4
Tune 4	2	3	2	7	1
Tune 5	3	5	6	1	0
Tune 6	9	4	2	0	0
Tune 7	4	9	0	2	0
Tune 8	0	3	4	2	6
Tune 9	0	6	4	3	2
Tune 10	1	4	8	2	0
Tune 11	3	4	5	3	0
Tune 12	1	4	4	5	1

This experiment obtained good evaluations for several tunes (e.g. Tune 1, 6, 7); on the other hand, it obtained relatively bad evaluations for other several tunes (e.g. Tune 3, 4, 8, 9, 12). Next section discusses the reasons of the results with the comments of examinees.

4.2 Feedback from Examinees

This section introduces free comments from examinees.

We asked examinees to give us any comments during the experiments introduced in Sections 4.1, and 4.2. Examinees commented as follows:

- Colors were dominant for the first impressions of pictures.
- Gradation of pictures was unremarkable because of the color arrangement.
- Some examinees might associate colors of music genre by fashion; for instance, colors of rock were black and white.

We think the above comments are key points to improve the evaluation of users. We will discuss about them more in the future.

We also asked examinees to give any comments on usability of MusCat. Many examinees gave us positive comments that MusCat was useful to use when they wanted to select tunes according to intuition or emotion, especially for unknown tunes. Some of them suggested us that MusCat can be an alternative of shuffle play mechanism of music player software.

We also got some constructive suggestions from examinees. Some of them suggested us to indicate the metadata or feature values of tunes they selected, even though they selected the tunes according to the impression of abstract pictures. We think it is interesting to add more effective pop-up mechanism for the selected tunes to indicate such information. Some other examinees commented that they might loss which part they are zooming in. We would like to add a navigation mechanism to solve the problem.

5. CONCLUSION AND FUTURE WORK

We presented MusCat, a music browser applying abstract pictures and zooming user interface. The technique uses features to cluster tunes, and to generate abstract pictures, so that users can recognize tunes more instantly and intuitionally without listening.

Following are our potential future work:

[Music feature:] Our current implementation calculates features from randomly selected 15-second segment of a tune. We would like to calculate features from all 15-second segments of a tune and select the most preferable or characteristic features from the calculation results.

[Abstract picture:] Our current abstract picture design is just an example, and therefore we think there may be better designs. Some of our examinees pointed that color is more important for the impression of pictures, than shapes and properties of objects. However, we have not yet found the best scheme to assign three colors to gradation, circles, and star. We need to discuss better schemes to assign the three colors. Another discussion is mood-based design of abstract pictures, since current design directly represents feature values.

[User Interface:] Current version of MusCat just plays the music by click operations, and simply indicates text information. We would like to extend the functionality. We would like to develop to show more metadata information of the selected tunes. Also, we would like to develop to play a set of tunes in the selected clusters by one click operation.

6. REFERENCES

[1] A. Gomi, R. Miyazaki, T. Itoh, J. Li: "CAT: A Hierarchical Image Browser Using a Rectangle Packing Technique," *12th International Conference on Information Visualization*, pp.82-87, 2008.

[2] K. Yamawaki, H. Shiizuka: "Characteristic recognition of the musical piece with correspondence analysis," *Journal of Kansei Engineering*, Vol. 7, No. 4, pp. 659–663, 2008.

[3] S. Kobayashi: *Color System*, Kodansha, Tokyo, 2001.

[4] M. Oda, T. Itoh, "MIST: A Music Icon Selection Technique Using Neural Network", *NICOGRAPH International,* 2007.

[5] P. Kolhoff, J. Preub, J. Loviscach: "Music Icons: Procedural Glyphs for Audio Files," *IEEE SIBGRAPI*, pp. 289-296, 2006.

[6] L. Lie, D. Liu, H. Zhang: "Automatic mood detection and tracking of music audio signals," *IEEE Transactions on Audio, Speech, and Language Processing,* Vol.14, pp. 5-18, 2006.

[7] M. Goto, T. Goto: Musicream: "New Music Playback Interface for Streaming, Sticking, Sorting, and Recalling Musical Pieces," *Proceedings of the 6th International Society for Music Information Retrieval*, pp.404-411, 2005.

[8] P. Lamere, D. Eck: "Using 3D Visualizations to explore and discover music," *Proceedings of the 8th International Society for Music Information Retrieval*, pp.173-174, 2007.

[9] Marsyas, http://marsyas.info/

[10] O. Lartillot: "MIRtoolbox," http://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mirtoolbox

[11] J. Harrison: *SYNAESTHESIA The Strangest Thing*, shin-yo-sha, 2006.

[12] RWC Music Database, http://staff.aist.go.jp/m.goto/RWC-MDB/

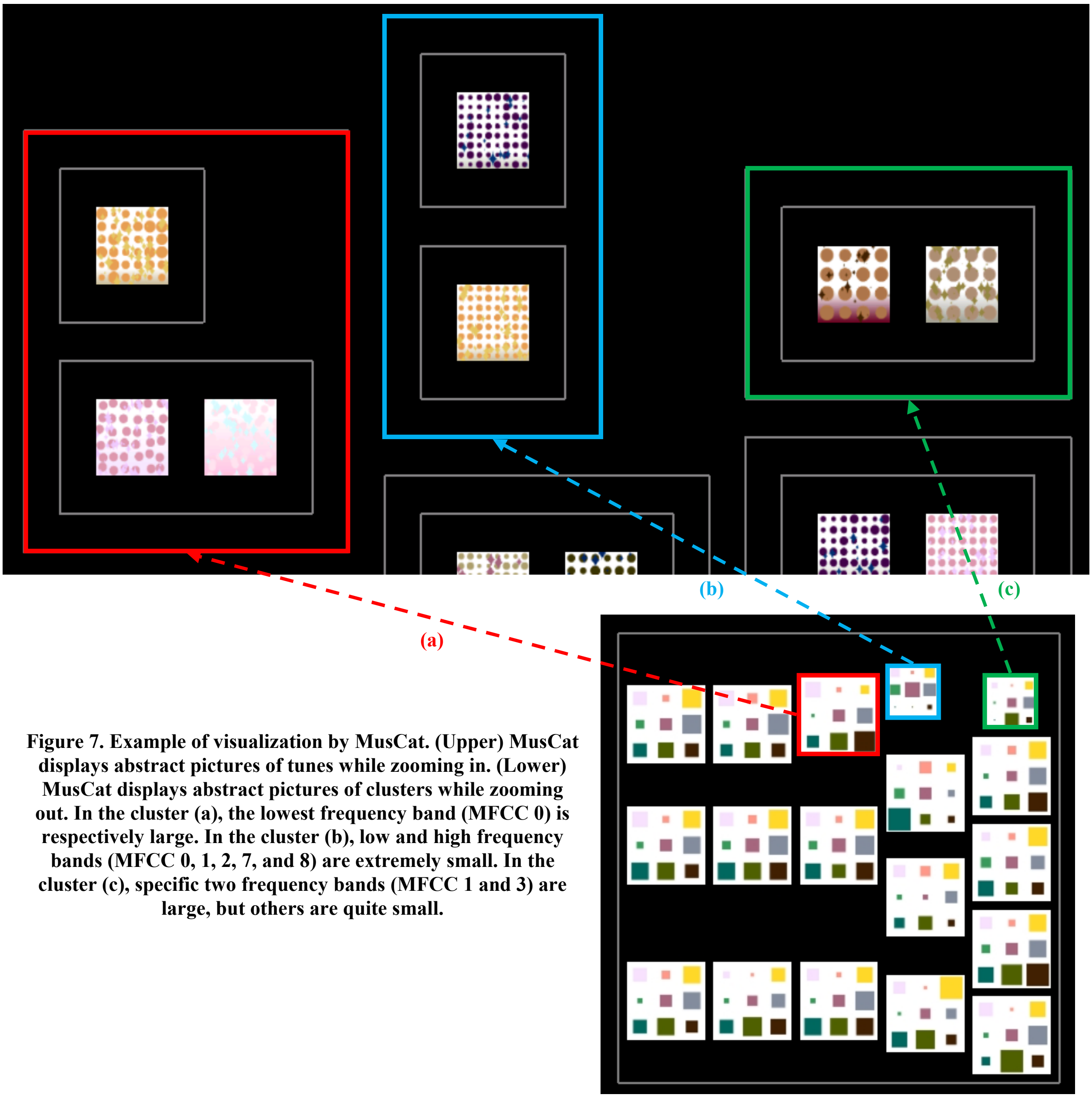

Figure 7. Example of visualization by MusCat. (Upper) MusCat displays abstract pictures of tunes while zooming in. (Lower) MusCat displays abstract pictures of clusters while zooming out. In the cluster (a), the lowest frequency band (MFCC 0) is respectively large. In the cluster (b), low and high frequency bands (MFCC 0, 1, 2, 7, and 8) are extremely small. In the cluster (c), specific two frequency bands (MFCC 1 and 3) are large, but others are quite small.