MUSIC THUMBNAILER: VISUALIZING MUSICAL PIECES IN THUMBNAIL IMAGES BASED ON ACOUSTIC FEATURES

Kazuyoshi Yoshii **Masataka Goto**
National Institute of Advanced Industrial Science and Technology (AIST)
{k.yoshii,m.goto}@aist.go.jp

ABSTRACT

This paper presents a principled method called *MusicThumbnailer* to transform musical pieces into visual thumbnail images based on acoustic features extracted from their audio signals. These thumbnails can help users immediately guess the musical contents of audio signals without trial listening. This method is consistent in ways that optimize thumbnails according to the characteristics of a target music collection. This means the appropriateness of transformation should be defined to eliminate ad hoc transformation rules. In this paper, we introduce three top-down criteria to improve memorability of thumbnails (generate gradations), deliver information more completely, and distinguish thumbnails more clearly. These criteria are mathematically implemented as minimization of brightness differences of adjacent pixels and maximization of brightness variances within and between thumbnails. The optimized parameters of a modified linear mapping model we assumed are obtained by minimizing a unified cost function based on the three criteria with a steepest descent method. Experimental results indicate that generated thumbnails can provide users with useful hints as to the musical contents of musical pieces.

1 INTRODUCTION

Music recommender systems are increasingly important in online music-distribution services to help users discover their favorite pieces among a huge music collection. For instance, recommender systems based on collaborative filtering [1, 2] recommend musical pieces to the user by taking into account someone else's ratings of those pieces. Content-based filtering systems [3, 4] select musical pieces that are similar to the user's favorites in terms of musical content (acoustic features). Recently, several hybrid systems that integrate these two techniques have been proposed to enable more accurate recommendations [5, 6].

An important problem that has not been resolved is that users cannot immediately grasp the musical contents of recommended pieces after these pieces are listed. Users have to listen to all listed pieces, including those they do not like, to find which pieces are worth listening to. This often prevents users from seamlessly listening to their favorite pieces. Worse still, trial listening is time-consuming because the information of audio signals (temporal media) is not simultaneously delivered to users whereas visual images (spatial media) can be easily skimmed through.

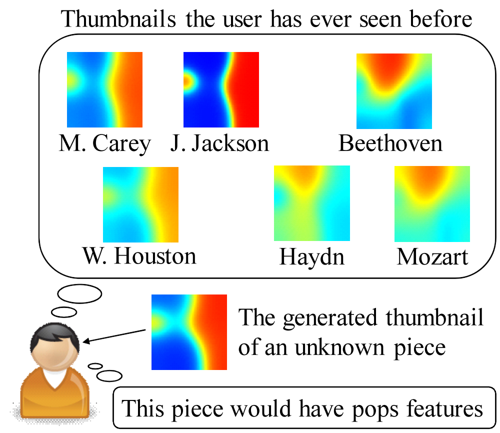

Figure 1. Expected scenario: A user can roughly guess the musical contents of unknown pieces by seeing only thumbnails and without time-consuming trial listening.

To solve this problem, we propose an audio-visual transformation method called *MusicThumbnailer* that generates compact images corresponding to the audio signals of individual pieces. This helps users guess the musical contents of audio signals without trial listening. For example, this method will work well in recommender systems, as shown in Fig. 1. Initially, users only glance at compact thumbnails attached to recommended pieces when they actually listen to these pieces. While accumulating this experience, users will unconsciously associate particular types of thumbnail with their preferred music. Users thus learn to understand the musical meanings of the thumbnails' features. Finally, users will be able to efficiently select audio signals of their desired pieces by using their eyes rather than their ears.

One advantage of our method is that the visual features (colors and patterns) of thumbnails are automatically optimized for a given collection. To achieve this, it is necessary to eliminate ad hoc rules that arbitrarily associate acoustic features with visual features, because these rules lack a principled justification that is consistent for different collections. In this paper, we define some top-down criteria on generated thumbnails from the viewpoint of usability, independently of the characteristics of music collections. We then mathematically represent these criteria in a unified cost function. Audio-visual associations are obtained in a self-organized way so that the cost function is minimized.

The rest of this paper is organized as follows. Section 2 introduces related work. Section 3 explains the principles of our audio-visual transformation method and its implementation. Section 4 reports on our experiment using the RWC Music Database [7]. Section 5 summarizes the key findings of this paper.

2 RELATED WORK

Many visualization methods have been proposed for spatial representation, organization, and browsing of music collections [8–16]. These methods typically locate musical pieces in a two- or three-dimensional space so that musical pieces that have similar musical contents (acoustic features) are arranged close to each other. This enables users to easily understand relationships between musical pieces because similarity in acoustic features can be observed as spatial distance. From a mathematical viewpoint, this kind of visualization can be interpreted as information compression of high-dimensional feature vectors according to some criteria. The self-organizing map (SOM) is often used for this purpose (e.g., Islands of Music [8]).

In a music-playback interface called Musicream [17], individual musical pieces are visualized as colored discs using an ad hoc rule. The disc color (hue and saturation) is determined from the color circle whose circumference and radius correspond to hue and saturation, respectively. Each piece is mapped into the circle according to its feature vector. Principal component analysis (PCA) is used to reduce the dimensionality of acoustic feature vectors to a two-dimensional vector on a plane. Musicovery [18] uses arbitrary rules for associating genres with colors specified in advance.

In the work described in this paper, we aimed to visualize individual pieces, rather than a collection, as compact images (thumbnails) without using ad hoc rules. In general, visual images are represented as super-high-dimensional vectors that contain the color values of numerous pixels. Therefore, our objective was to find an appropriate mapping from a low-dimensional acoustic space (several-tens dim.) to a high-dimensional visual space (several-thousands dim.). A unique feature of this problem lies in this drastic increase in degrees of freedom. To solve such an ill-posed problem by using an optimization method, it is necessary to incorporate some criteria on the appropriateness of the mapping.

3 MUSIC THUMBNAILER

This section describes our method of generating thumbnails of musical pieces based on acoustic features.

3.1 Problem Specification

Given a collection of musical pieces (audio signals) as input data, our objective is to output appropriate thumbnails (visual images) that reflect acoustic features extracted from these pieces. We first prepare some criteria to evaluate the appropriateness of the generated thumbnails as discussed later. In this paper, we focus on generating gray-scale thumbnails as the first step towards obtaining full-color thumbnails in the future. This means we have only to deal with the brightness of pixels contained in each thumbnail.

We first define constant values in advance. Let N be the number of musical pieces. Let S be the number of acoustic features taken into account. Let T be the number of pixels contained in each thumbnail, where T is the product of width, W, and height, H; i.e., $T = WH$.

The input data is given by $X = [\boldsymbol{x}_1 \boldsymbol{x}_2 \cdots \boldsymbol{x}_N]$, which is a collection of feature vectors extracted from all musical pieces. Here, $\boldsymbol{x}_n = (x_{n,1}, \cdots, x_{n,S})^T$ is an S-dimensional feature vector of piece n $(1 \leq n \leq N)$, where $x_{n,s}$ $(1 \leq s \leq S)$ represents the value of feature s in piece n.

The output data is represented as $Y = [\boldsymbol{y}_1 \boldsymbol{y}_2 \cdots \boldsymbol{y}_N]$, which is a set of brightness vectors of generated thumbnails. $\boldsymbol{y}_n = (y_{n,1,1}, \cdots, y_{n,1,H},\ y_{n,2,1}, \cdots, y_{n,2,H},\ \cdots,\ y_{n,W,1}, \cdots, y_{n,W,H})^T$ is a T-dimensional brightness vector of piece n, where $y_{n,w,h}$ $(1 \leq w \leq W, 1 \leq h \leq H)$ is the brightness of pixel (w, h) in thumbnail n. The range of $y_{n,w,h}$ is given by $0 < y_{n,w,h} < 1$.

3.2 Top-down Criteria for Principled Transformation

To evaluate the appropriateness of the generated thumbnails, we introduce three top-down criteria from the viewpoint of user friendliness (usability) as follows:

1. *Memorability:* Each thumbnail should be easily remembered by users. The visual pattern is an important factor that affects the ease of remembering thumbnails, as shown in Fig. 2. This is related to how easily users can understand the musical meanings of visual patterns based on their memories. We assume that gradation images are suitable to our purpose.

2. *Informational delivery:* Each thumbnail should provide a large amount of information to users. This is achieved if thumbnail's pixels have a wide variety of brightness, as shown in Fig. 3. This enables users to associate a thumbnail with detailed information about the music content.

3. *Distinguishability:* Users should be able to easily distinguish thumbnail images of different pieces. To enable this, each thumbnail should have a distinctive visual pattern that explicitly reflects the content of music, as shown in Fig. 4. This enables users to efficiently find their favorite pieces based on thumbnails they recognize.

Note that these criteria mention neither specific genres of music nor specific colors of thumbnails.

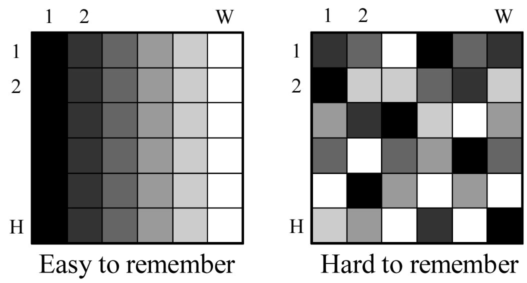

Figure 2. Difference in ease of remembering thumbnails. Both thumbnails have the same histogram of brightness.

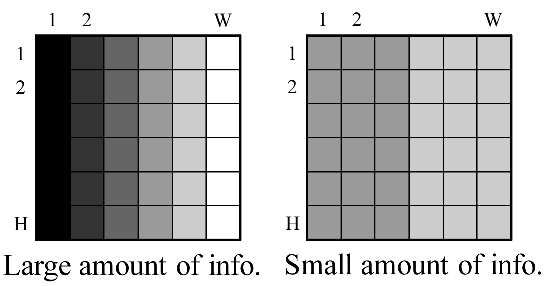

Figure 3. Difference of the amount of information obtained from thumbnails.

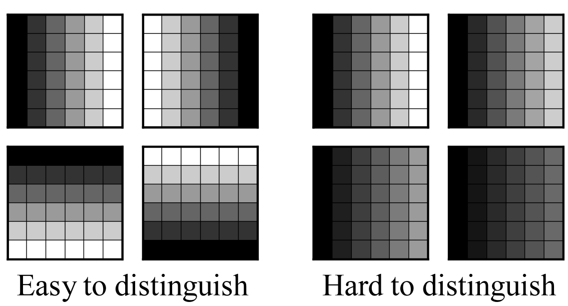

Figure 4. Difference in ease of distinguishing thumbnails of different pieces.

These top-down criteria enable us to design thumbnails in a non-ad-hoc way. In general, system designers tend to define ad hoc rules that directly associate musical contents of musical audio signals with specific colors of visual images (e.g., rock-red, jazz-green, and classic-blue). However, the appropriateness of these arbitrary rules cannot be guaranteed. In contrast, the criteria we proposed only regulate the appropriateness of transformation. The actual colors and patterns of thumbnails are optimized for a given collection in a self-organized way so that the criteria are best satisfied. Therefore, our approach is methodologically and mathematically sound.

A remaining problem here is how to mathematically implement these criteria. In this paper, we will present a simple implementation of the three criteria below.

3.3 Mathematical Formulation

From a mathematical point of view, the objective is to obtain optimal parameters of a mapping model that transforms a S-dimensional space to a T-dimensional space so that a cost function defined according to the three criteria is minimized. This transformation has a high degree of freedom; a higher-dimensional space can always preserve the complete information of an original space. We thus incorporate linear mapping into a mapping model. This is reasonable because linear mapping can strongly limit the degree of freedom in transformation. In addition, musical pieces which are close to each other in an acoustic space are expected to be mapped still close to each other in a visual space. This effect is suitable to achieve the expected scenario shown in Fig. 1. We define the mapping model as

$$Y = \textit{Sig}(AX), \tag{1}$$

where A is a T-by-S transformation matrix to be optimized, which is given by

$$\begin{bmatrix} \boldsymbol{A}_{1,1}^T \\ \vdots \\ \boldsymbol{A}_{1,H}^T \\ \vdots \\ \boldsymbol{A}_{W,1}^T \\ \vdots \\ \boldsymbol{A}_{W,H}^T \end{bmatrix} = \begin{bmatrix} A_{1,1,1} & A_{1,1,2} & \cdots & A_{1,1,S} \\ \vdots & & & \vdots \\ A_{1,H,1} & A_{1,H,2} & \cdots & A_{1,H,S} \\ \vdots & & & \vdots \\ A_{W,1,1} & A_{W,1,2} & \cdots & A_{W,1,S} \\ \vdots & & & \vdots \\ A_{W,H,1} & A_{W,H,2} & \cdots & A_{W,H,S} \end{bmatrix} \tag{2}$$

and *Sig* is the sigmoid function given by

$$\textit{Sig}(x) = \frac{1}{1+e^{-x}} \quad (-\infty < x < \infty). \tag{3}$$

In Eq. (1), we applied the sigmoid function to each cell of matrix AX so that the value of each cell of matrix Y ranges from 0 to 1. Note that the differential of the sigmoid function is given by $\textit{Sig}'(x) = \textit{Sig}(x)\,(1 - \textit{Sig}(x))$.

To evaluate the appropriateness of the mapping model, we define a cost function, C, as

$$C = C_s + \alpha_w C_w + \alpha_b C_b, \tag{4}$$

where C_s, C_w, and C_b are the costs corresponding to the three criteria described in Section 3.2, and α_w and α_b are the weighting parameters. We mathematically define these three costs below.

3.3.1 Minimization of adjacent-pixel distances

To generate gradation images, we focus on a necessary condition that the brightness values of adjacent pixels should be close to each other. One way to mathematically implement this condition is to minimize the differences of the brightness values of adjacent pixels included in each thumbnail.

However, this calculation is not efficient because it would have to be repeated for all thumbnails. To solve this problem, we directly define the cost function, C_s, for the transformation matrix, A, as follows:

$$C_s = \frac{1}{WHS} \sum_{w,h,s} D_{w,h,s}, \tag{5}$$

where $D_{w,h,s}$ is the average of the following eight distances:

$$\begin{aligned} D_{w,h,s} &= \frac{1}{8} \sum_{i,j=\pm 1} \left(A_{w,h,s} - A_{w+i,h+j,s}\right)^2 && (6)\\ &+ \frac{1}{8} \sum_{i=\pm 1} \left(A_{w,h,s} - A_{w+i,h,s}\right)^2 && (7)\\ &+ \frac{1}{8} \sum_{j=\pm 1} \left(A_{w,h,s} - A_{w,h+j,s}\right)^2 . && (8) \end{aligned}$$

3.3.2 *Maximization of within-thumbnail variances*

To increase the amount of information delivered by a thumbnail, the brightness variance within the thumbnail should be maximized. We formulate this condition for each thumbnail and define the cost function, C_w, based on the average of brightness variances over all thumbnails:

$$C_w = -\frac{1}{N} \sum_n \frac{1}{WH} \sum_{w,h} \left(y_{n,w,h} - \bar{y}_n\right)^2, \tag{9}$$

where $\bar{y}_n$ is an average of the brightness values of wh pixels within a thumbnail of piece n, given by

$$\bar{y}_n = \frac{1}{WH} \sum_{w,h} y_{n,w,h}. \tag{10}$$

Note that maximization of within-thumbnail variances is equivalent to minimization of the cost function C_w.

3.3.3 *Maximization of between-thumbnail variances*

To enable users to clearly distinguish generated thumbnails, the brightness vectors of these thumbnails should be far from each other. A simple way to satisfy this condition is to maximize the brightness variance between all thumbnails, where the between-thumbnail variance is calculated in each position, (w, h). We define the cost function, C_b, based on the average of brightness variances over all positions as

$$C_b = -\frac{1}{WH} \sum_{w,h} \frac{1}{N} \sum_n \left(y_{n,w,h} - \bar{y}_{w,h}\right)^2, \tag{11}$$

where $\bar{y}_{w,h}$ is an average of the brightness values of n pixels in the same position (w, h), given by

$$\bar{y}_{w,h} = \frac{1}{N} \sum_n y_{n,w,h}. \tag{12}$$

Note that maximization of between-thumbnail variances is equivalent to minimization of the cost function C_b.

3.4 Parameter Optimization

To minimize the total cost function, C, as an optimization method, we use a steepest descent method that iteratively updates the parameters until the cost reduction is converged. The updating formula is given by

$$A_{w,h,s} \leftarrow A_{w,h,s} - \eta \frac{\partial C}{\partial A_{w,h,s}}, \tag{13}$$

where η is a learning parameter ($0 < \eta < 1$) and $\frac{\partial C}{\partial A_{w,h,s}}$ is decomposed into three terms:

$$\frac{\partial C}{\partial A_{w,h,s}} = \frac{\partial C_s}{\partial A_{w,h,s}} + \alpha_w \frac{\partial C_w}{\partial A_{w,h,s}} + \alpha_b \frac{\partial C_b}{\partial A_{w,h,s}}. \tag{14}$$

3.4.1 *Derivation of updating formula*

We now explain how to derive the updating formula. The first term in Eq. (14) is obtained by

$$\frac{\partial C_s}{\partial A_{w,h,s}} = \frac{2}{WHS} \left(A_{w,h,s} - \bar{A}_{w,h,s}\right), \tag{15}$$

where $\bar{A}_{w,h,s}$ is an average of the values in the vicinity of $A_{w,h,s}$, given by

$$\bar{A}_{w,h,s} = \frac{A_{w,h\pm 1,s} + A_{w\pm 1,h,s} + A_{w\pm 1,h\pm 1,s}}{8}. \tag{16}$$

The second and third terms are calculated as

$$\begin{aligned} \frac{\partial C_w}{\partial A_{w,h,s}} &= \frac{\partial C_w}{\partial y_{n,w,h}} \cdot \frac{\partial y_{n,w,h}}{\partial A_{w,h,s}}, && (17)\\ \frac{\partial C_b}{\partial A_{w,h,s}} &= \frac{\partial C_b}{\partial y_{n,w,h}} \cdot \frac{\partial y_{n,w,h}}{\partial A_{w,h,s}}, && (18) \end{aligned}$$

where

$$\begin{aligned} \frac{\partial C_w}{\partial y_{n,w,h}} &= -\frac{2}{NWH} \left(y_{n,w,h} - \bar{y}_n\right), && (19)\\ \frac{\partial C_b}{\partial y_{n,w,h}} &= -\frac{2}{NWH} \left(y_{n,w,h} - \bar{y}_{w,h}\right), && (20)\\ \frac{\partial y_{n,w,h}}{\partial A_{w,h,s}} &= \textit{Sig'}(\boldsymbol{A}_{h,w}^T \boldsymbol{x}_n) x_{n,s} && (21)\\ &= \textit{Sig}(\boldsymbol{A}_{h,w}^T \boldsymbol{x}_n) \Big(1 - \textit{Sig}(\boldsymbol{A}_{h,w}^T \boldsymbol{x}_n)\Big) x_{n,s}. && (22) \end{aligned}$$

3.4.2 *Visual effects of updating formula*

Next, we will discuss the visual effects of the three terms of Eq. (14) in the mathematically derived updating formula. Eq. (15) means the first term tries to make transformation coefficients close to their smoothed versions. Eq. (16) corresponds to a visual processing algorithm that is used for smoothing images by using a convolution matrix. Eq. (19) and Eq. (20) mean the second and third terms try to make the brightness value of each pixel far from the within- and between-thumbnail averages. This enhances the dynamic range of each thumbnail and the variety of generated thumbnails. These effects intuitively match our expectation.

4 EXPERIMENT

This section reports on a thumbnail-generation experiment done to evaluate the usefulness of MusicThumbnailer.

4.1 Experimental Conditions

As a music collection, we used the “RWC Music Database: Music Genre” (RWC-MDB-G-2001) [7], which consists of 100 pieces ($N = 100$) in total with three pieces prepared for each of 33 genres and one for a cappella. This database is divided into 10 main genre categories (popular, rock, dance, jazz, Latin, classical, march, world, vocal, and traditional Japanese music) and 33 subcategories (pops, ballad, rock, heavy metal, rap/hip-hop, house, techno, funk, soul/R&B, big band, modern jazz, fusion, bossa nova, samba, reggae, tango, baroque, classic, romantic, modern, brass band, blues, folk, country, gospel, African, Indian, flamenco, chanson, canzone, traditional-style Japanese popular music, Japanese folk music, and ancient Japanese court music).

To extract acoustic features, we used the MARSYAS [19]. We obtained a 42-dimensional feature vector for each piece, which consists of the average and variance of local spectral features (centroid, rolloff, and flux) and zero-crossings across the entire piece (8 dimensions), average and variance of Mel-frequency cepstral coefficients (MFCC) across the entire piece (26 dimensions), and rhythmic content features reflecting periodicity in beat (8 dimensions). We then used PCA to reduce the dimensionality; the 42-dimensional space was transformed into a 20-dimensional space still accounting for 95% of the variance of the original data ($S = 20$).

The size of thumbnails was 50×50 ($W = H = 50, T = 2500$). In this experiment, gray-scale thumbnails were converted to full-color ones according to a lookup table shown in Table 1 (called “Jet” color scheme in MATLAB) from an aesthetic point of view. Note that this is just for convenience because we should essentially eliminate such an ad hoc rule. The values of parameters, α_w, α_b, and η were empirically set to 0.4, 0.4, and 0.1.

4.2 Experimental Results

The experimental results indicate that the generated thumbnails are useful for guessing the musical contents, as shown in Fig. 5. At the main-category level, thumbnails of popular/rock pieces, those of classical (orchestra) pieces, and those of marches seem to be close to each other, respectively. In the dance category, similar thumbnails were obtained in each subcategory, where we can find the similarity between rap/hip-hop and funk. The thumbnails of funk pieces somewhat resemble those of reggae and African pieces across the main category. This corresponds with the fact that funk music has been developed while absorbing the characteristics of African music and reggae. In the categories of Latin, world, and traditional Japanese, we can roughly say that similar thumbnails were obtained in each subcategory. In the vocal category, each subcategory yielded the similar thumbnails, which have especially unique patterns among the database. In the jazz category, however, there were comparatively wide variations in thumbnails of each category. Moreover, fusion pieces tend to reflect and mix the styles of other pieces.

Table 1. Lookup table for converting brightness to RGB.

Brightness	0.00	0.33	0.66	1.00
RGB	(0,0,1)	(0,1,1)	(1,1,0)	(1,0,0)

5 CONCLUSION

We presented an audio-visual transformation method called MusicThumbnailer that generates thumbnail images reflecting acoustic features of audio signals. To achieve principled transformation free from ad hoc rules, we designed three top-down criteria regarding memorability, informational delivery, and distinguishability. These criteria are used to evaluate the appropriateness of generated thumbnails from the viewpoint of usability rather than to associate specific acoustic features with actual colors and patterns of thumbnails. From a mathematical viewpoint, we formulated this problem as a constrained minimization of a cost function based on the three criteria. The experiment showed promising results as to the usefulness of MusicThumbnailer.

Many issues still remain regarding the refinement of our method through subjective experiments. First, we plan to introduce a new criterion to preserve the topological relations of feature vectors in audio-visual transformation. Then, we will improve the mathematical implementation of each criterion and attempt to use a more sophisticated optimization algorithm that can achieve fast convergence while avoiding the local-minimum problem. Several experiments using different features and collections would be important. An interesting application of our method would be to generate a visual effect (a temporal sequence of visual images) that dynamically represents the local musical contents in a musical piece. This can be done by interpreting feature vectors extracted from individual time frames in a musical piece in the same way as for those extracted from individual pieces in a music collection. Such a visualizer will give users a practical overview of structures within a musical piece.

6 REFERENCES

[1] Shardanand, U. and Maes, P., “Social Information Filtering: Algorithms for Automating “Word of Mouth”,” *ACM Conf. on Human Factors in Computing Systems*, 1995, pp. 210–217.

[2] Cohen, W. and Fan, W., “Web-Collaborative Filtering: Recommending Music by Crawling the Web,” *Computer Networks*, Vol. 33, No. 1–6, pp. 685–698, 2000.

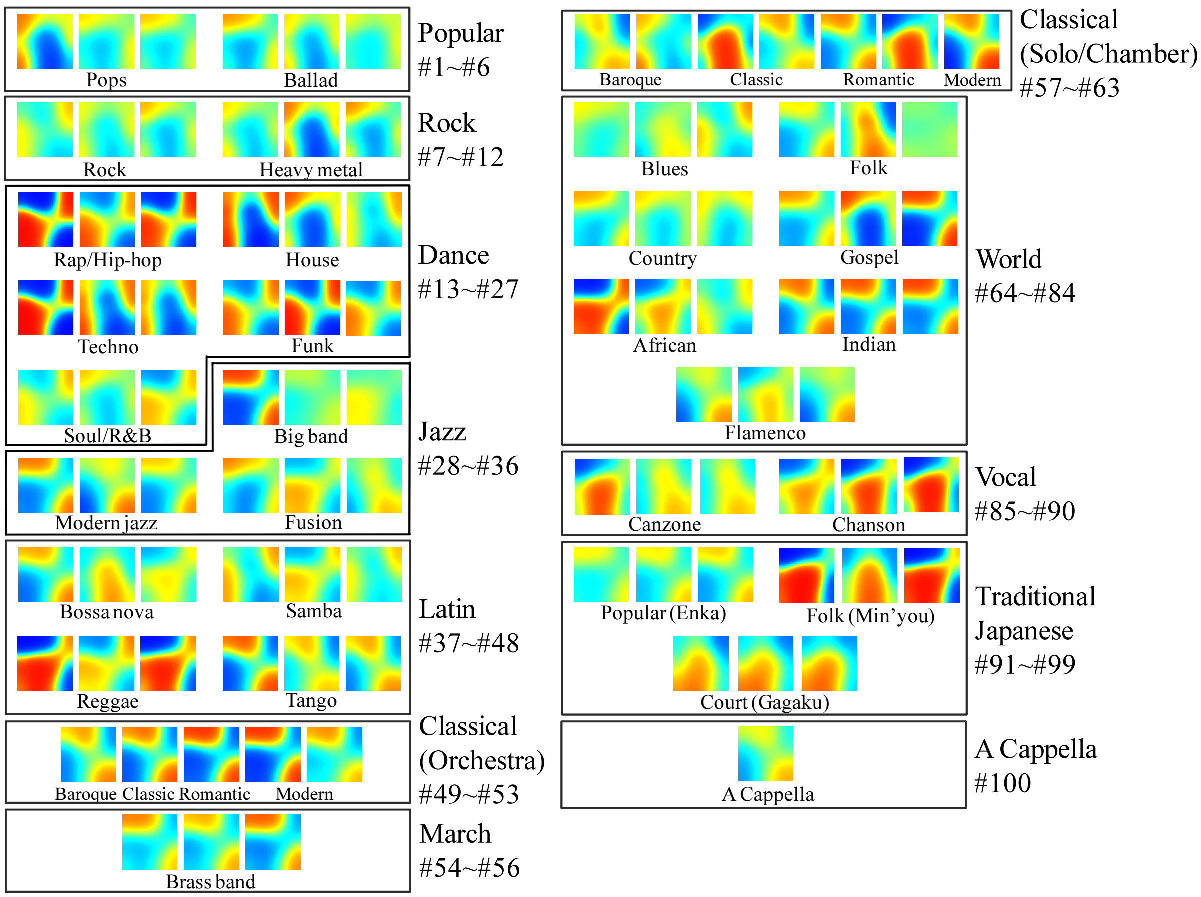

Figure 5. Experimental results of generating thumbnail images of 100 pieces included in RWC-MDB-G-2001 [7].

[3] Hoashi, K., Matsumoto, K., and Inoue, N., "Personalization of User Profiles for Content-based Music Retrieval based on Relevance Feedback," *ACM Multimedia*, 2003, pp. 110–119.

[4] Logan, B., "Music Recommendation from Song Sets," *ISMIR*, 2004, pp. 425–428.

[5] Celma, O., Ramirez, M., and Herrera, P., "Foafing the Music: A Music Recommendation System based on RSS Feeds and User Preferences," *ISMIR*, 2005, pp. 464–457.

[6] Yoshii, K., Goto, M., Komatani, K., Ogata, T., and Okuno, H. G., "An Efficient Hybrid Music Recommender System Using an Incrementally-trainable Probabilistic Generative Model," *IEEE Trans. on Audio, Speech and Language Processing*, Vol. 16, No. 2, pp. 435–447, 2008.

[7] Goto, M., Hashiguchi, H., Nishimura, T., and Oka, R., "RWC Music Database: Music Genre Database and Musical Instrument Sound Database," *ISMIR*, 2005, pp. 229-230.

[8] Pampalk, E., Dixon, S., and Widmer, G., "Exploring Music Collections by Browsing Different Views," *ISMIR*, Vol. 28, Mo. 2, pp. 49–62, 2004.

[9] Torrens, M., Hertzog, P., and Arcos, J., "Visualizing and Exploring Personal Music Libraries," *ISMIR*, 2004, pp. 421–424.

[10] Mörchen, F., Ultsch, A., Nöcker, M., and Stamm, C., "Databionic Visualization of Music Collections According to Perceptual Distances," *ISMIR*, 2005, pp. 396–403.

[11] Mayer, R., Dittenbach, M., and Rauber, A., "PlaySOM and PocketSOMPlayer: Alternative Interfaces to Large Music Collections," *ISMIR*, 2005, pp. 618–623.

[12] Mayer, R., Lidy, T., Rauber, A., "The Map of Mozart," *ISMIR*, 2006, pp. 351–352.

[13] Knees, P., Schedl, M., Pohle, T., and Widmer. G., "An Innovative Three-dimensional User Interface for Exploring Music Collections Enriched with Meta-Information from the Web," *ACM Multimedia*, 2006, pp. 17–24.

[14] Leitich, S. and Topf, M., "Globe of Music: Music Library Visualization Using GEOSOM," *ISMIR*, 2007, pp. 167–170.

[15] Donaldson, J. and Knopke, I., "Music Recommendation Mapping and Interface based on Structual Network Entropy," *ISMIR*, 2007, pp. 181–182.

[16] Lamere, P. and Eck, D., "Using 3D Visualizations to Explore and Discover Music," *ISMIR*, 2007, pp. 173–174.

[17] Goto, M. and Goto, T., "Musicream: New Music Playback Interface for Streaming, Sticking, Sorting, and Recalling Musical Pieces," *ISMIR*, 2005, pp. 404–411.

[18] Musicovery: http://musicovery.com/.

[19] Tzanetakis, G. and Cook, P., "MARSYAS: A Framework for Audio Analysis," *Organized Sound*, No. 4, Vol. 3, pp. 169–175, 2000.