A HISTOGRAM DENSITY MODELING APPROACH TO MUSIC EMOTION RECOGNITION

Ju-Chiang Wang[1,2]*, Hsin-Min Wang*[2]*, and Gert Lanckriet*[1]

[1] Department of Electrical and Computer Engineering, UC San Diego
[2] Institute of Information Science, Academia Sinica

ABSTRACT

Music emotion recognition is concerned with developing predictive models that comprehend the affective content of musical signals. Recently, a growing number of attempts has been made to model the music emotion as a probability distribution in the valence-arousal (VA) space to better account for the subjectivity. In this paper, we present a novel histogram density modeling approach that models the emotion distribution by a 2-D histogram over the quantized VA space and learns a set of latent histograms to predict the emotion probability density of a song from audio. The proposed model is free from parametric distribution assumptions over the VA space, easy to implement, and extremely fast to train. We also extend our model to deal with the temporal dynamics of time-varying emotion labels. Comprehensive performance study on two larger-scale datasets demonstrates that our approach achieves comparable performance to the state-of-the-art ones, but with much better training and testing efficiency.

Index Terms— Affective computing, music information retrieval, subjectivity, temporal dynamics, emotion tracking

1. INTRODUCTION

Recent years have witnessed increasing research interests in automatic music emotion recognition (MER), which holds the promise of managing the ever increasing volume of digital music in a content-based way [1, 2]. In a usual setting, a MER model is trained by machine learning to capture the mapping between musical features and emotion, and then the performance can be measured by the deviation between the predicted and ground truth emotions of a song [3–7].

Despite that considerable research has been undertaken in the past few years, MER still remains challenging. This can be attributed to two fundamental issues in modeling the music emotion: First, the perceived emotion in music is by nature subjective – it is typically highly dependent on the listener and the situational context [8–10]. Therefore, a *deterministic* approach that associates music with a single label might not work well in practice [11]. Second, the affective content of music changes temporally as the music evolves [12–16]. For better performance, it is desirable to consider the dynamic relationships between music and emotion.

To handle the subjectivity issue, a growing number of attempts has been made to model the music emotion as a *parametric* probability distribution in the valence-arousal (VA) space [12, 17–19]. The underlying approaches typically assume that the VA emotion annotations from different listeners can be modeled by a bi-variate Gaussian distribution. While this permits analytical tractability, there is lack of consensus in the literature on whether the VA space is indeed Euclidean,[1] and the parameter estimation could be biased if the annotation samples of a song are insufficient. An intuitive solution is to quantize the VA space into a $G \times G$ grid and train a predictive model for each cell independently [19]. Such approach is usually referred to as *heatmap* [14].

To tackle the problem of modeling the temporal dynamics of musical emotion, on the other hand, several sophisticated approaches have been proposed, such as Kalman filtering [13], Conditional Random Fields (CRF) [14], Continuous Conditional Random Fields (CCRF) [15], and Continuous Conditional Neural Fields (CCNF) [16]. Although notable progress has been made, relatively little effort has focused on optimizing the training efficiency scalable to a larger dataset. Moreover, the concept of time-varying emotion tracking lends itself to applications such as visualizing emotion in sync with music playback [1, 21]. It is thus important for a method to predict the emotion on local music signals in real time.

In this paper, we propose the *histogram density mixture* (HDM) model, a novel probabilistic model that is interpretable and theoretically sound. The HDM-based approach models the emotion distribution of a song using 2-dimensional histogram density estimation over the $G \times G$-quantized VA space (cf. Fig. 1) and learns a mixture of latent histograms, each of which is associated with an audio topic. It makes emotion prediction on unseen audio by linearly combining the learned latent histograms with the weights generated from different audio topics. Our proposed model is free from parametric distribution assumptions over the VA space, easy to implement with the EM algorithm [22], extend-

This work was supported by Academia Sinica–UCSD Postdoctoral Fellowship to Ju-Chiang Wang.

[1] One obvious problem is that the VA space in a typical interface for user to make the valence and arousal ratings is bounded, e.g. the one in [20] is bounded by [-1,1]. This may result in density discontinuity when modeling the annotation density around the boundaries.

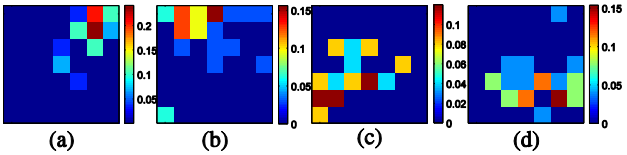

Fig. 1. The histogram density distributions of the emotion annotations of four 30-second music excerpts in the VA space:[2] (a) *Splish Splash* by Bobby Darin, (b) *Dick Johnson* by Pussy Galore, (c) *American Gothic* by David Ackles, and (d) *Pledging My Love* by Johnny Ace.

able to handle the temporal dynamics of emotion annotations, and extremely fast to train, and predicts emotion efficiently.

We conduct performance study on two emotion annotated datasets, *AMG1608* and *MTurk*. AMG1608 [23] is a newly complied dataset containing the emotion labels of 1,608 30-second music excerpts annotated by 665 listeners. While, MTurk [24] is a widely used dataset for testing the performance of moment-by-moment emotion tracking. Our empirical result demonstrates that the proposed HDM approach achieves comparable performance to the state-of-the-art ones, but with much better training and testing efficiency. For reproductivity, the Matlab codes for implementing and evaluating HDM have been made publicly available.[3] One can easily train a HDM model on 3,600 training instances within 2 second using a laptop computer.

2. THE HDM MODEL

2.1. Histogram density estimation for music emotion

To account for the subjectivity nature of emotion perception, each song is typically annotated by multiple subjects. Let $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_U]$ denote the set of annotations of a song, where $\mathbf{y}_u \in \mathbb{R}^2$ is the individual annotation of the u-th listener, and U is the number of annotations available for the song. Our approach starts with partitioning each emotion dimension equally into G bins and obtain a $G \times G$ grid representation of the VA space. Let $\mathbf{H} \in \mathbb{R}^{G \times G}$ denote the 2-D histogram density for the grid, where $h(i,j)$ denotes the probability of the (i,j)-th cell of $\mathbf{H}$. We count the number of annotations in $\mathbf{Y}$ falling in the (i,j)-th cell as initial $h(i,j)$, and then normalize $\mathbf{H}$ by U such that $\sum_i \sum_j h(i,j) = 1$. Fig. 1 shows four examples of $\mathbf{H}$.

2.2. Learning the latent histograms of HDM

Suppose we have a labeled dataset $\mathcal{L} = \{(\mathbf{H}_n, \mathbf{x}_n)\}_{n=1}^N$, where $\mathbf{H}_n$ is the histogram density matrix summarizing the annotations of the n-th song, and $\mathbf{x}_n = [x_{n1}, \ldots, x_{nK}]$, $\sum_k x_{nk} = 1$, is a probability vector that captures the acoustic features of the song. Note that each probability x_{nk} in $\mathbf{x}_n$ is generated based on a specific audio topic a_k. We will detail this process later in Section 3. Then, our goal is to learn K number of latent histograms $\mathcal{H} = \{\mathbf{\Phi}_k\}_{k=1}^K$, such that each $\mathbf{\Phi}_k \in \mathbb{R}^{G \times G}$ is associated with an audio topic a_k.

Let $h_n(i,j)$ and $\phi_k(i,j)$ denote the (i,j)-th element of $\mathbf{H}_n$ and $\mathbf{\Phi}_k$, respectively. We fit the HDM model by maximizing the log-likelihood of $\mathcal{H}$ on $\mathcal{L}$ defined as follows:

$$\log p(\mathcal{H} \mid \mathcal{L}) = \sum_n \sum_{i,j} h_n(i,j) \log \sum_k x_{nk} \phi_k(i,j) \,. \quad (1)$$

To maximize $\log p(\mathcal{H}|\mathcal{L})$ with respect to $\mathcal{H}$, we apply the EM algorithm [22]. In the E-step, we compute

$$\gamma(k,n,i,j) \leftarrow \frac{x_{nk}\phi_k(i,j)}{\sum_l x_{nl}\phi_l(i,j)} \,. \quad (2)$$

In the M-step, we update $\mathcal{H}$ by

$$\phi'_k(i,j) \leftarrow \frac{\sum_n \gamma(k,n,i,j) h_n(i,j)}{\sum_n \sum_q \sum_r \gamma(k,n,q,r) h_n(q,r)} \,. \quad (3)$$

2.3. Modeling the temporal dynamics of emotion

In the case of modeling the time-varying music emotion, emotion labels are annotated on local music signals consecutively over time. Let $[(\mathbf{H}_n^{(1)}, \mathbf{x}_n^{(1)}), \ldots, (\mathbf{H}_n^{(T_n)}, \mathbf{x}_n^{(T_n)})]$ denote the sequence of data tuples for the n-th song, where T_n is the length of the song. Our goal is to jointly model the relationship between $\mathbf{x}_n^{(t)}$ and its locally windowed emotion histograms $[\mathbf{H}_n^{(t-\tau)}, \ldots, \mathbf{H}_n^{(t)}, \ldots, \mathbf{H}_n^{(t+\tau)}]$, and to learn 2τ+1 models $\{\mathcal{H}^{(r)}\}_{r=1}^{2\tau+1}$, where $\mathcal{H}^{(r)} = \{\mathbf{\Phi}_k^{(r)}\}_{k=1}^K$.

Let $h_n^{(t)}(i,j)$ and $\phi_k^{(r)}(i,j)$ denote the (i,j)-th element of $\mathbf{H}_n^{(t)}$ and $\mathbf{\Phi}_k^{(r)}$, respectively. We derive the log-likelihood by

$$\log p(\mathcal{H}^{(1)}, \ldots, \mathcal{H}^{(2\tau+1)} \mid \mathcal{L}, W) = \sum_n \sum_{t=1}^{T_n} \sum_{v=-\tau}^{\tau} w^{(v)} \sum_{i,j} h_n^{(t+v)}(i,j) \log \sum_k x_{nk}^{(t)} \phi_k^{(v+\tau+1)}(i,j) \,, \quad (4)$$

where the weights $W = [w^{(-\tau)}, \ldots, w^{(0)}, \ldots, w^{(\tau)}]$ are set to descend from its center according to the power law: $w^{(0)}$=1, $w^{(\pm 1)}$=0.5, $w^{(\pm 2)}$=0.25, and so on. We refer to this modified model as HDM dynamic (HDMd). The optimization problem of Eq. 4 can be divided into 2τ+1 sub-problems, each is in turn solved by the EM algorithm (cf. Eqs. 2 and 3).

2.4. Emotion prediction

To make prediction with HDM, we can estimate the emotion histogram density based on the probability vector $\hat{\mathbf{x}} = [\hat{x}_1, \ldots, \hat{x}_K]$ computed from the acoustic features of an unseen song:

$$\hat{\mathbf{H}} = \sum_k \hat{x}_k \mathbf{\Phi}_k \,. \quad (5)$$

[2] The VA space is ranging in between [-1,1] and quantized with $G = 7$. Its horizontal and vertical axes correspond to valence and arousal, respectively.

[3] https://github.com/asriverwang/HDM_codes

We can further compute the centroid of $\hat{\mathbf{H}}$ to represent the mean of the emotion prediction for simplicity:

$$\boldsymbol{\mu} = \sum\nolimits_{i,j} \hat{h}(i,j)\bar{\mathbf{y}}(i,j)\,, \tag{6}$$

where $\bar{\mathbf{y}}(i,j)$ is the VA coordinate values of the (i,j)-th cell's center. For time-varying emotion prediction with HDMd, given a $\hat{\mathbf{x}}^{(t)}$ we compute the corresponding 2τ+1 histograms, $\{\hat{\mathbf{H}}_v^{(t)}\}_{v=-\tau}^{\tau}$. Then, we re-estimate the histogram density at t by the weighted combination of the results predicted on its adjacent music signals $\{\hat{\mathbf{x}}^{(t+v)}\}_{v=-\tau}^{\tau}$:

$$\hat{\mathbf{H}}^{(t)} = \frac{1}{\sum_v w^{(v)}} \sum_{v=-\tau}^{\tau} w^{(v)} \hat{\mathbf{H}}_v^{(t+v)}\,. \tag{7}$$

3. ACOUSTIC FEATURE REPRESENTATION

Suppose that a song contains a sequence of acoustic feature vectors computed over windowed frames of its audio signal, we apply a K-component Gaussian mixture model (GMM) to encode the acoustic feature vectors of the song into a probability vector $\mathbf{x}$. Specifically, a GMM is pre-trained using the EM algorithm in an unsupervised manner on a large collection of acoustic feature vectors without emotion annotations [18]. By assigning each component of the GMM as an audio topic a_k, we can compute a K-dimensional posterior probability vector that represents how likely the acoustic features of a frame is characterized by each audio topic [22]. As a song comprises multiple frames, we average the frame-level posterior probability vectors over the whole song to obtain $\mathbf{x}$. Note that one can easily extend the frame-level features to block-level ones to compute the posterior probability vectors [25], so that more local temporal characteristics can be captured.

4. EXPERIMENTS

4.1. Dataset

We adopt the AMG1608 and MTurk datasets for evaluating song-level MER and second-by-second emotion tracking, respectively. AMG1608 contains 1608 30-second music excerpts annotated by 665 subjects, and each excerpt is annotated by 15–32 subjects. During the song selection stage, the mood category information from All Music Guide (AMG) and a tag2VA algorithm [26] are applied to ensure that the emotions of the selected excerpts are well-balanced in the four quadrants of the VA space. Then, Amazon Mechanical Turk (AMT), an online crowdsourcing engine, is exploited to collect the emotion annotations, following previous work [3, 14]. More details can be found in [23].

MTurk [24] is composed of 240 pieces of 15-second music clips drawn from the uspop2002 database. Each clip is annotated by 7 to 23 subjects via AMT. Each subject is asked to rate the VA values of 11 randomly selected music clips on a per-second basis using a graphical interface. The VA rating scale is normalized to [–0.5, 0.5] for making meaningful comparison with the result reported in [14].

4.2. Acoustic features

For AMG1608, we employ MIRToolbox [27] and YAAFE [28] to extract the frame-based acoustic features with a frame size of 50ms and 50% overlap. These features include MFCC-related (static, delta, and delta-delta MFCCs), tonal, spectral, and temporal features, leading to a 72-dimensional vector for a frame [23]. Then, we use the block-level representation [25] that consists of 16 consecutive frames, and each block overlaps with its previous one by 12 frames. A block-level feature vector is generated by concatenating the mean and standard deviation of the frame-based feature vectors. The block-level feature vectors are used to train the GMM and to compute the posterior probability vector $\mathbf{x}$.

For MTurk, we adopt the frame-based MFCCs (20-D) and spectral contrast (14-D) features provided by the authors of [24]. As the provided audio feature vectors contain only static features, we compute the delta- and delta-delta- features over the sequence of the static ones [29]. These dynamic features are concatenated to the static ones to add the information of the temporal dynamics of the audio signals. Since each emotion histogram is aligned with a specific one-second clip of a song, we extract the feature vectors from each one-second clip. We use the frame-level (instead of block-level) feature vectors to train the GMM and to compute $\mathbf{x}^{(t)}$, as the audio signal for a $\mathbf{x}^{(t)}$ is short (i.e., 1 sec).

4.3. Qualitative analysis of the learned models

We depict the learned latent histograms on the entire AMG1608 with K=32 in Fig. 2, from which two observations are made. First, each latent histogram explains the semantic meaning of the corresponding audio topic. For example, T15, T16, and T26 are highly associated with the first quadrant of the VA space (e.g., happy, delighted topics), whereas T6, T19, and T32 are strongly correlated to the second (e.g., angry), third (e.g., sad), and fourth (e.g., calm) quadrants of the VA space, respectively. Second, one can qualitatively assess the quality of a latent topic. Some topics might be too vague to be useful for robust emotion modeling, such as T20 and T30. For better performance, some future study can be done to downweight or remove such latent topics accordingly.

4.4. Experimental result and discussion

We stop the EM learning when the increase ratio of log-likelihood (cf. Eqs. 1 and 4) is smaller than 0.0001. We run our experiments on twelve cores of a server with a 2.66 GHz XEON X5650 CPU running Matlab R2011b on a 64-bit operating system. The average computing time on each train-test process is reported.

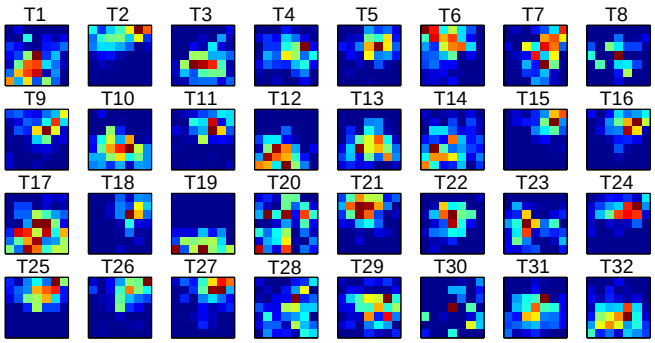

Fig. 2. The latent histogram topics (K=32) on AMG1608.

Table 1. Performance comparison on AMG1608.

Approach	ED	R^2 Val.	R^2 Aro.	Time
SVR-RBF [19]	0.2895	0.1409	0.6613	20 min
AEG [18]	0.2869	0.1579	0.6686	15 min
HDM (G=5)	0.2887	0.1419	0.6624	0.3 sec
HDM (G=7)	0.2879	0.1513	0.6652	0.3 sec
HDM (G=10)	0.2899	0.1490	0.6589	0.4 sec

We perform 3-fold cross-validation on AMG1608 and use two metrics to measure the performance: the Euclidean distance (ED) and R^2 (the coefficient of determination) [1] between the predicted VA values (i.e., $\boldsymbol{\mu}$) and the ground truth ones (the average VA ratings across the listeners of a song). ED is the smaller the better, while the opposite holds for R^2. We compare HDM with support vector regression (SVR) [30] and acoustic emotion Gaussians (AEG) [18]. The former, which stands for the baseline, is implemented by LIBSVM [31] along with RBF kernel and a grid parameter search to optimize its performance. The latter, which represents the state-of-the-art approach on AMG1608, uses the same setting as that of HDM to compute the acoustic feature representation of $\mathbf{x}$.[4] According to the cross-validation on the training set, we determine K=256 for both HDM and AEG.

Table 1 presents the result for AMG1608. We also show the performance of HDM with different G values. From the result, HDM (G=7) significantly outperforms (p-value<5%) the baseline (SVR-RBF) in all metrics, and achieves fairly competitive performance against AEG with much lower computing time (0.3 sec vs. 900 sec), which is 3000 times faster than AEG. HDM with different Gs performs the best at G=7, indicating that an overly simplified or complicated histogram quantization may reduce the capability of emotion modeling.

For MTruk, we follow the experiment protocol in [14] to avoid the "album-effect" while splitting the 240 songs into 70% and 30% for training and testing, respectively, and repeat the validation 10 times with different train-test distributions. The performance is measured by ED and the root-mean-square (RMS) error [16]. Smaller RMS leads to better performance. We use K=128, G=7, and τ=3 for HDM(d).

[4] AEG applies a bivariate GMM to model the emotion distribution in the VA space and can be viewed as the parametric, continuous version of HDM.

Table 2. Comparison of different HDM(d) settings on MTurk.

Setting	ED	RMS Val.	RMS Aro.
HDMd (dyn-Contr.)	**0.1222**	0.2036	**0.1920**
HDM (dyn-Contr.)	0.1230	0.2058	0.1948
HDMd (stat-Contr.)	0.1252	0.2060	0.1984
HDM (stat-Contr.)	0.1284	0.2114	0.2088
HDMd (dyn-MFCCs)	0.1256	**0.2026**	0.2006
HDM (dyn-MFCCs)	0.1272	0.2044	0.2016
HDMd (stat-MFCCs)	0.1281	0.2090	0.2046
HDM (stat-MFCCs)	0.1308	0.2146	0.2120

Table 3. Performance comparison on MTurk.

Approach	ED	RMS Val.	RMS Aro.	Time
SVR-RBF [19]	0.132	0.220	0.208	42 min
CRF [14]	0.122	–	–	>11 hr
CCRF [15]	0.136	0.223	0.204	–
CCNF [16]	**0.116**	0.205	**0.166**	>2 hr
AEG [18]	0.128	0.206	0.202	35 min
HDMd	0.122	**0.204**	0.192	**1.4 sec**

Table 2 shows the performance comparison of HDMd and HDM with different settings on MTurk, where "Contr." stands for the spectral contrast feature, "dyn-" means using the concatenation of static, delta, and delta-delta features, and "stat-" uses only the static features. Three observations can be made. First, HDMd consistently outperforms HDM (which makes independent prediction on each $\hat{\mathbf{x}}^{(t)}$), suggesting that our proposed method for modeling the temporal dynamics of emotion is effective. Second, "dyn-" consistently outperforms "stat-" regardless of any settings. This demonstrates the importance of using dynamic acoustic features in a codebook-like approach [32] (in our case, GMM). Third, MFCC features perform better in predicting the valence.

In Table 3, we compare HDMd with other emotion tracking approaches on MTurk. For SVR, we follow the same setting in the AMG1608 case. AEG is trained with K=128 and predicts on each $\hat{\mathbf{x}}^{(t)}$ independently. We report the performance of CRF in [14], and that of CCRF and CCNF in [16]. Note that the experimental settings for CCRF and CCNF [16] may be slightly different from ours. In general, the performance of HDMd is comparable to its competitors, but HDMd spends much lower computing time on training and testing. In particular, the superior performance of HDMd in RMS valence is remarkable. Such observation suggests HDMd a promising approach, as it is typically more difficult to model the valence perception from audio signals [1].

5. CONCLUSION

We have presented a novel probabilistic approach, coined as the histogram density mixture model, to model the relationship between emotion and music. We have also extended the model to handle the temporal dynamics of music emotion. Our experimental result has demonstrated its effectiveness and outstanding efficiency in training and prediction.

6. REFERENCES

[1] Y.-H. Yang and H. H. Chen, *Music Emotion Recognition*, CRC Press, 2011.

[2] Y. E. Kim, E. M. Schmidt, R. Migneco, B. G. Morton, P. Richardson, J. J. Scott, J. A. Speck, and D. Turnbull, "Music emotion recognition: A state of the art review," in *Proc. Int. Soc. Music Information Retrieval Conf.*, 2010, pp. 255–266.

[3] T. Eerola, O. Lartillot, and P. Toiviainen, "Prediction of multidimensional emotional ratings in music from audio using multivariate regression models," in *Proc. Int. Soc. Music Information Retrieval Conf.*, 2009, pp. 621–626.

[4] A. Huq, J. P. Bello, and R. Rowe, "Automated music emotion recognition: A systematic evaluation," *J. New Music Res.*, vol. 39, no. 3, pp. 227–244, 2010.

[5] E. Schubert, "Modeling perceived emotion with continuous musical features," *Music Perception*, vol. 21, no. 4, pp. 561–585, 2004.

[6] L. Lu, D. Liu, and H. Zhang, "Automatic mood detection and tracking of music audio signals," *IEEE Trans. Audio, Speech & Lang. Proc.*, vol. 14, no. 1, pp. 5–18, 2006.

[7] M. Soleymani, M. N. Caro, E. Schmidt, C.-Y. Sha, and Y.-H. Yang, "1000 songs for emotional analysis of music," in *Proc. Int. Workshop on Crowdsourcing for Multimedia*, 2013, pp. 1–6.

[8] M. Zentner and K. R. Scherer, "Emotional effects of music: production rules," in *Music and Emotion: Theory and Research*, P. N. Juslin and J. A. Sloboda, Eds. Oxford University Press, New York, 2001.

[9] K. R. Scherer, "Which emotions can be induced by music? what are the underlying mechanisms? and how can we measure them," *J. New Music Res.*, vol. 33, no. 5, pp. 239–251, 2004.

[10] J. Hanratty, *Individual and situational differences in emotional expression*, Ph.D. thesis, School of Psychology, Queen's University Belfast,, 2010.

[11] Y.-H. Yang, C. C. Liu, and H. H. Chen, "Music emotion classification: A fuzzy approach," in *Proc. ACM Multimedia*, 2006, pp. 81–84.

[12] E. M. Schmidt and Y. E. Kim, "Prediction of time-varying musical mood distributions from audio," in *Proc. Int. Soc. Music Information Retrieval Conf.*, 2010, pp. 465–470.

[13] E. M. Schmidt and Y. E. Kim, "Prediction of time-varying musical mood distributions using Kalman filtering," in *Proc. Int. Conf. Machine Learning and Applications*, 2010, pp. 655–660.

[14] E. M. Schmidt and Y. E. Kim, "Modeling musical emotion dynamics with conditional random fields," in *Proc. Int. Soc. Music Information Retrieval Conf.*, 2011, pp. 777–782.

[15] V. Imbrasaitė, T. Baltrušaitis, and P. Robinson, "Emotion tracking in music using continuous conditional random fields and baseline feature representation," in *Proc. Int. Workshop on Affective Analysis in Multimeda*, 2013.

[16] V. Imbrasaitė, T. Baltrušaitis, and P. Robinson, "CCNF for continuous emotion tracking in music: Comparison with CCRF and relative feature representation," in *Proc. IEEE Int. Conf. Multimedia and Expo Workshops*, 2014.

[17] J. A. Russell, "A circumplex model of affect," *J. Personality and Social Science*, vol. 39, no. 6, pp. 1161–1178, 1980.

[18] J.-C. Wang, Y.-H. Yang, H.-M. Wang, and S.-K. Jeng, "The acoustic emotion Gaussians model for emotion-based music annotation and retrieval," in *Proc. ACM Multimedia*, 2012, pp. 89–98.

[19] Y.-H. Yang and H. H. Chen, "Prediction of the distribution of perceived music emotions using discrete samples," *IEEE Trans. Audio, Speech and Lang. Proc.*, vol. 19, no. 7, pp. 2184–2196, 2011.

[20] Y.-H. Yang, Y.-F. Su, Y.-C. Lin, and H. H. Chen, "Music emotion recognition: The role of individuality," in *Proc. ACM Int. Workshop on Human-Centered Multimedia*, 2007, pp. 13–21.

[21] J.-C. Wang, H.-M. Wang, and S.-K. Jeng, "Playing with tagging: A real-time tagging music player," in *Proc. IEEE Int. Conf. Acoustics, Speech & Signal Processing*, 2012, pp. 77–80.

[22] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer-Verlag New York, Inc., 2006.

[23] Yu-An Chen, Y.-H. Yang, J.-C. Wang, and H. H. Chen, "The AMG1608 dataset for music emotion recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech & Signal Processing*, 2015, [Online] `http://amg1608.blogspot.tw/`.

[24] J. A. Speck, E. M. Schmidt, B. G. Morton, and Y. E. Kim, "A comparative study of collaborative vs. traditional musical mood annotation," in *Proc. Int. Soc. Music Information Retrieval Conf.*, 2011, pp. 549–554.

[25] K. Seyerlehner, *Content-Based Music Recommender Systems: Beyond simple Frame-Level Audio Similarity*, Ph.D. thesis, Johannes Kepler University Linz, 2010.

[26] J.-C. Wang, Y.-H. Yang, K. Chang, H.-M. Wang, and S.-K. Jeng, "Exploring the relationship between categorical and dimensional emotion semantics of music," in *Proc. Int. ACM workshop on Music Information Retrieval with User-centered and Multimodal Strategies*, 2012, pp. 63–68.

[27] O. Lartillot and P. Toiviainen, "A Matlab toolbox for musical feature extraction from audio," in *Proc. Int. Conf. Digital Audio Effects*, 2007, pp. 237–244.

[28] B. Mathieu, S. Essid, T. Fillon, J. Prado, and G. Richard, "YAAFE, an easy to use and efficient audio feature extraction software.," in *Proc. Int. Soc. Music Information Retrieval Conf.*, 2010, pp. 441–446.

[29] S. Furui, "Speaker-independent isolated word recognition based on emphasized spectral dynamics," in *Proc. IEEE Int. Conf. Acoustics, Speech & Signal Processing*, 1986, vol. 11, pp. 1991–1994.

[30] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A regression approach to music emotion recognition," *IEEE Trans. Audio, Speech & Lang. Proc.*, vol. 16, no. 2, pp. 448–457, 2008.

[31] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[32] Y. Vaizman, B. McFee, and G. Lanckriet, "Codebook-based audio feature representation for music information retrieval," *IEEE/ACM Trans. Audio, Speech, & Language Processing*, vol. 22, no. 10, pp. 1483–1493, 2014.