EMOTION TRACKING IN MUSIC USING CONTINUOUS CONDITIONAL RANDOM FIELDS AND RELATIVE FEATURE REPRESENTATION

Vaiva Imbrasaitė, Tadas Baltrušaitis, Peter Robinson

Computer Laboratory
University of Cambridge
{Vaiva.Imbrasaite, Tadas.Baltrusaitis, Peter.Robinson}@cl.cam.ac.uk

ABSTRACT

Digitization of how people acquire music calls for better music information retrieval techniques, and dimensional emotion tracking is increasingly seen as an attractive approach. Unfortunately, the majority of models we still use are borrowed from other problems that do not suit emotion prediction well, as most of them tend to ignore the temporal dynamics present in music and/or the continuous nature of Arousal-Valence space. In this paper we propose the use of Continuous Conditional Random Fields for dimensional emotion tracking and a novel feature vector representation technique. Both approaches result in a substantial improvement on both root-mean-squared error and correlation, for both short and long term measurements. In addition, they can both be easily extended to multimodal approaches to music emotion recognition.

Index Terms— Arousal-Valence space, continuous emotions, machine learning, feature representation, acoustic features

1. INTRODUCTION

Most of music singles and a substantial (and growing) proportion of albums are sold in their digital versions in UK[1]. With a fifth of all the consumers having fully transitioned to digital music, the need for more intelligent and powerful ways of managing digital music libraries is stronger than ever.

There is no doubt that people associate emotions with music and use emotion related terms when searching for music [2]. This naturally leads to a conclusion that we need an efficient and accurate way of automatically inferring emotion in large collections of songs. There is a growing body of research that attempts to tackle this problem, but the majority of work is still focused on trying to assign categorical emotion labels to classical music, while leaving emotion tracking and popular music in general largely unattended.

In this paper we introduce an adaptation to continuous conditional random fields (CCRF), that has so far never been used in emotion tracking in music (section 3.2). We also explore a novel way of representing acoustic features for machine learning (section 3.1) and show how these two techniques on their own and together can improve the accuracy of emotion tracking (section 4).

The code for CCRF and test-scripts that would allow easy reproduction of results is available on our website[1].

2. RELATED WORK

Both dimensional emotion representation using Arousal-Valence (AV) space and emotion tracking rather than predicting emotion for the entire sequence is gaining popularity not only in the field of music and emotion, but also in other areas of affective computing.

While there has been some work done on trying to combine emotion classification and emotion tracking (Schubert *et al.* [3] explored music emotion labeling using discrete emotion faces; Lu *et al.* [4] proposed to segment a song into ranges of stable emotion and do emotion classification on them), the majority of work focuses on emotion tracking in the AV space.

Within the approaches of emotion tracking, a large part of research has been focused on trying to infer the emotion label over a time window independently of the surrounding music (bag-of-frames approach) (Korhonen *et al.* [5], Panda and Paiva [6], Schmidt and Kim [7], Schmidt *et al.* [8], etc.). This approach is obviously limited, as it fails to acknowledge and exploit the temporal properties of music. So another solution is to incorporate temporal information in the feature vector either by using features extracted over varying window length for each second/sample [9], or by using machine learning techniques that are adapted for sequential learning (e.g. sequential stacking algorithm used by Carvalho and Chao [10], Kalman filtering or Conditional Random Fields (CRF) used by Schmidt and Kim [11, 12]). Interestingly, it has also been reported [8, 6] that taking the average of the time-varying emotion produces results that are statistically significantly better than simply performing emotion recognition on the whole piece of music.

[1] http://www.cl.cam.ac.uk/research/rainbow/projects/ccrf/

Dimensional emotion tracking in videos tends to use similar, more complex machine learning techniques. Nicolaou *et al.* [13] propose the use of Output-Associative Relevance Vector Machine (OA-RVM) for dimensional and continuous prediction of emotions based on automatically tracked facial feature points, by employing a window that covers a set of past and future outputs. Another approach proposed by Nicolaou *et al.* [14] that exploits temporal characteristics of emotion prediction. It uses bidirectional Long Short-Term Memory neural networks, which enable the model to use the information from the whole sequence of frames.

Similarly to Schmidt and Kim [12], the work done by Wöllmer *et al.* [15] uses CRF for discrete emotion recognition by quantizing the continuous labels for valence and arousal based on a selection of acoustic features. The main disadvantage of that is the loss of relationship between the different quantized bins, which could be better exploited by using a continuous approach rather than a discretized one.

3. METHODOLOGY

In this section we describe the two techniques that we introduce to emotion tracking in music (sections 3.1 and 3.2) and explain our experimental design and describe the dataset we are using.

3.1. Relative feature vector representation

There is a strong belief in the field of music and emotion that expectancy is a very important factor in our experience of listening to music. It is believed that violation of, or conformity to expectancy when listening to music is a (main) source of musical emotion (proven by studies in neuroimaging [16], experimental aesthetics [17], etc.). We therefore hypothesise that changing the focus of features we extract from music from their absolute values to more song-centered values would make a positive effect on our models.

The approach we take replaces absolute feature values with relative ones. We calculate the average over a song for each audio feature, include that in our feature vector and represent each feature as a difference between its (absolute) value at that time step and the average over that song (which we will refer to as the relative representation). We want to make sure that any possible improvement does not come simply from the addition of the average of a feature. To do this we also construct feature vectors with the original absolute feature values together with their averages for the song.

3.2. CCRF

We want to model the affect continuously rather than turning this problem into a classification one by discretising the signal as done by many previous approaches [18]. We want to model the temporal relationships between each time step, since emotion has temporal properties and is not instantaneous. A recent and promising approach that would allow us to model such temporal relationships is the Continuous Conditional Random Fields [19] (CCRF). It is an extension of the classic Conditional Random Fields [20] (CRF) to the continuous case. Furthermore, it has been recently extended so it can be used for continuous emotion prediction, incorporating temporal information [21].

3.3. Model definition

CCRF is an undirected graphical model where conditional probability $P(y|x)$ is modeled explicitly. It is a discriminative approach, which has shown promising results for sequence labeling and segmentation [22]. This is in contrast to generative models where a joint distribution $P(y,x)$ is modeled instead. The graphical model that represents our CCRF for emotion prediction is shown in Figure 1.

In our discussion we will use the following notation: $\{\mathbf{x}_1^{(q)}, \mathbf{x}_2^{(q)}, \ldots, \mathbf{x}_n^{(q)}\}$ is a set of observed input variables (in our case an SVR prediction), $\{y_1^{(q)}, y_2^{(q)}, \ldots, y_n^{(q)}\}$ is a set of output variables that we wish to predict, $\mathbf{x}_i^{(q)} \in \mathcal{R}^m$ and $y_i^{(q)} \in \mathcal{R}$, n is the number of frames/time-steps in a sequence, m is the number of predictors used (in our case we just use one, but multiple predictions per modality can be easily used), q indicates the q^{th} sequence of interest. When there is no ambiguity, q is omitted for clarity.

Our CCRF model for a particular sequence is a conditional probability distribution with the probability density function:

$$P(\mathbf{y}|\mathbf{X}) = \frac{\exp(\Psi)}{\int_{-\infty}^{\infty} \exp(\Psi) d\mathbf{y}} \tag{1}$$

$$\Psi = \sum_i \sum_{k=1}^{m} \alpha_k f_k(y_i, \mathbf{X}) + \sum_{i,j} \beta g(y_i, y_j, \mathbf{X}) \tag{2}$$

Above $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ is the set of input feature vectors (can be represented as a matrix with per frame observations as rows), $\mathbf{y} = \{y_1, y_2, \ldots, y_n\}$ is the unobserved variable. $\int_{-\infty}^{\infty} \exp(\Psi) d\mathbf{y}$ is the normalisation (partition) function which makes the probability distribution a valid one (by making it sum to 1). Following the convention of Qin *et al.* [19] we call f vertex features, and g edge features (in our model we use a single vertex and a single edge feature, so we drop the k in some further equations). The model parameters α, and β would be provided for inference and need to be estimated during learning.

3.4. Feature functions

We define two types of features for our CCRF model, vertex features f_k and edge feature g.

$$f_k(y_i, \mathbf{X}) = -(y_i - \mathbf{X}_{i,k})^2, \tag{3}$$

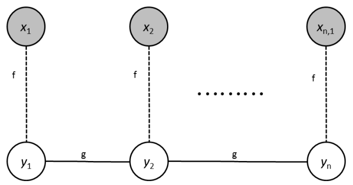

Fig. 1. Graphical representation of the CCRF model. x_i represents the the i^{th} observation, and y_i is the unobserved variable we want to predict. Dashed lines represent the connection of observed to unobserved variables (f is the vertex feature). The solid lines show connections between the unobserved variables (edge features).

$$g(y_i, y_j, \mathbf{X}) = -\frac{1}{2} S_{i,j} (y_i - y_j)^2. \tag{4}$$

Vertex features f_k represent the dependency between the $\mathbf{X}_{i,k}$ and y_i, for example dependency between a static emotion prediction from a regressor and the actual emotion label. Intuitively, the corresponding α_k for vertex feature f_k represents the reliability of that particular predictor. In our work we use a single predictor, however, it is possible to use multiple regressors [21].

Edge feature g represents the dependency between observations y_i and y_j, for example how related is the emotion prediction at time step j to the one at time step i. This is also affected by the similarity measure S. Because we are using a fully connected model, the similarity S allows us to control the strength or existence of such connections. In our work we use the following similarity:

$$S_{i,j} = \begin{cases} 1, & |i-j| = 1 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Thus we connect neighboring observations. The framework allows for easy creation of different similarity measures which could be appropriate for other applications.

The learning phase of CCRF will determine the parameters α and β. For example, it can learn that for one emotion neighbor similarities are more important than for others.

Same as Radosavljevic *et al.* [23], Qin *et al.* [19] and Baltrušaitis *et al.* [21], our feature function models the square error between prediction and a feature. Therefore the elements of the feature vector $\mathbf{x}_i$ should be predicting the unobserved variable y_i. This can be achieved using Support Vector Regression used in our work.

3.5. Learning

In this section we describe how to estimate the parameters $\{\alpha, \beta\}$ of a CCRF with quadratic vertex and edge functions. We are given training data $\{\mathbf{x}^{(q)}, \mathbf{y}^{(q)}\}_{q=1}^{M}$ of M sequences, where each $\mathbf{x}^{(q)} = \{\mathbf{x}_1^{(q)}, \mathbf{x}_2^{(q)}, \dots, \mathbf{x}_n^{(q)}\}$ is a sequence of inputs and each $\mathbf{y}^{(q)} = \{y_1^{(q)}, y_2^{(q)}, \dots, y_n^{(q)}\}$ is a sequence of real valued outputs. We also use the matrix $\mathbf{X}$ to denote the concatenated sequence of inputs.

In learning we want to pick the α and β values that optimise the conditional log-likelihood of the CCRF:

$$L(\alpha, \beta) = \sum_{q=1}^{M} \log P(\mathbf{y}^{(q)} | \mathbf{x}^{(q)}) \tag{6}$$

$$(\bar{\alpha}, \bar{\beta}) = \arg\max_{\alpha,\beta} (L(\alpha, \beta)) \tag{7}$$

As the problem is convex [19], the optimal parameter values can be determined using standard techniques such as stochastic gradient ascent, or other general optimisation techniques.

In order to guarantee that our partition function is integrable we constrain $\alpha > 0$ and $\beta > 0$ [19, 23]. Such constrained optimisation can be achieved by using partial derivatives with respect to $\log \alpha$ and $\log \beta$ instead of just α and β. We also add a regularisation term in order to avoid overfitting. The regularisation is controlled by λ_α and λ_β hyper-parameters (determined during cross-validation).

$$\frac{\partial \log(P(\mathbf{y}|\mathbf{X}))}{\partial \log \alpha} = \alpha\left(\frac{\partial \log(P(\mathbf{y}|\mathbf{X}))}{\partial \alpha} - \lambda_\alpha \alpha\right) \tag{8}$$

$$\frac{\partial \log(P(\mathbf{y}|\mathbf{X}))}{\partial \log \beta} = \beta\left(\frac{\partial \log(P(\mathbf{y}|\mathbf{X}))}{\partial \beta} - \lambda_\beta \beta\right) \tag{9}$$

The derivation and definition of the partial derivatives can be found in Baltrušaitis *et al.* [21].

The full learning algorithm is described in Algorithm 1.

Algorithm 1 Our CCRF learning algorithm

```
Require: $\{\mathbf{X}^{(q)}, \mathbf{y}^{(q)}, S_q\}_{q=1}^{M}$
  Params: number of iterations T, learning rate $\nu$, $\lambda_\alpha, \lambda_\beta$
  Initialise parameters $\{\alpha, \beta\}$
  for r = 1 to T do
    for i = 1 to N do
      Compute gradients of current query (Eqs.(8),(9))
      $\log \alpha = \log \alpha + \nu \frac{\partial \log(P(\mathbf{y}|\mathbf{X}))}{\partial \log \alpha}$
      $\log \beta = \log \beta + \nu \frac{\partial \log(P(\mathbf{y}|\mathbf{x}))}{\partial \log \beta}$
      Update $\{\alpha, \beta\}$
    end for
  end for
  return $\{\bar{\alpha}, \bar{\beta}\} = \{\alpha, \beta\}$
```

3.6. Inference

Because our CCRF model can be viewed as a multivariate Gaussian, inferring $\mathbf{y}$ values that maximise $P(\mathbf{y}|\mathbf{x})$ is straightforward. The prediction is the mean value of the distribution.

$$\mathbf{y}' = \arg\max_{\mathbf{y}} (P(\mathbf{y}|\mathbf{X})) \tag{10}$$

For more details on the inference algorithm please see Baltrušaitis *et al.* [21].

3.7. Dataset

The dataset that we have used in our experiments is, to our knowledge, the only publicly available emotion tracking dataset of music extracts labeled on the arousal-valence dimensional space. The data [24] has been collected using Mechanical Turk (MTurk)[2], asking paid participants to label 15-second long excerpts with continuous emotion ratings on the AV space, with another 15 seconds given as a practice for each song. The songs in the dataset cover a wide range of genres—pop, various types of rock, hip-hop/rap, etc, and are drawn from the "uspop2002"[3] database containing popular songs. The dataset consists of 240 15-second clips (without the practice run) with 16.9 ± 2.7 ratings for each clip. In addition, the dataset contains a standard set of features extracted from those musical clips: MFCCs, octave-based spectral contrast, statistical spectrum descriptors, chromagram and a set of EchoNest[4] features.

We average these labels on a second by second basis and use the average value for each second as our ground truth. The original labels show a reasonable amount of agreement between different participants, but the variance is still rather large. Interestingly, the variance for both the arousal and valences axes is the same, which, compared to the difference in results achieved by state-of-the-art models, implies that there is something substantial that they are still missing.

3.8. Design of the experiments

For our experiments, we work with 1s long frames, as that is the resolution of the labels provided in the MTurk dataset. We use the non-EchoNest features provided in the dataset (MFCCs, octave-based spectral contrast, statistical spectrum descriptors and chromagram) averaged over 1s period. We use Support Vector Regression (SVR) as the baseline method (and the input to CCRF) since it is one of the most popular machine learning techniques used in the field. For SVR-based experiments we use a bag-of-frames approach, where we create a feature vector for each second of a song and encode no relationship to other feature vectors. The CCRF-based approach still works on vectors for each second of a song, but it inherently contains some information about their temporal relationship with each other.

We train a separate (SVR or CCRF) model for each axis and use both linear and RBF kernels for our SVR-based experiment (and provide the results for both). CCRF uses the predictions of the best performing SVR model for that particular axis as input.

Throughout all of the experiments we employ careful cross-validation techniques in order to minimize overfitting as much as possible. This also ensures that different approaches are exposed to the same training and testing data in each fold.

"Album effect" is now a widely recognized issue in the field of music emotion recognition. It has been reported and accepted that the so called "album effect" can artificially improve the performance as machine learning models overfit to a particular set of post production techniques used on an album [25]. It is thefore, in general, worth making sure that songs from the same album are all within a single fold. For the dataset we are using, however, removing the album effect did not make any difference to the results, when tested on SVR. We suspect that the reason for that is that a large majority of songs come from unique albums—the 240 songs we are using come from 200 different albums. For this reason, we decided to simplify the cross-validation and only make sure that all of the samples from a song (and not necessarily from the same album) are within the same fold.

3.8.1. Cross-validation

The experiments we run can be split into two parts—SVR and CCRF. The experimental design for them is slightly different, but based on the same core idea—we use 5-fold cross-validation to produce the final results, and 2-fold cross-validation for the training of our machine learning methods.

For SVR-based experiments, we split the whole dataset into two parts– 4/5 for training and 1/5 for testing. We then use 2-fold cross-validation (splitting it into two equal parts) on the training set to learn the hyper-parameters of the SVR model, which we then use for training on the whole training set. This process is repeated 5 times and the results are averaged over the 5 folds.

The process for the CCRF-based experiments contains an extra step. We use the same 5-folds as in the SVR-based experiments. We then split the 4/5 training dataset into two parts—one for SVR and one for CCRF, and perform 2-fold cross-validation on them to learn the hyper-parameters in the same way we do for the SVR-based experiments.

4. RESULTS

For each experiment that we run, we calculate 4 evaluation measures—correlation (corr) and root-mean-square error (RMS) that are calculated over the whole testing set and averaged over the 5 folds (which we refer to as long measures), and correlation and RMS that are calculated over the whole song for each song in the fold, and averaged over all of the songs (which we refer to as short measures). The long term correlation is squared, as that is the common way of representing correlation in the field, while the short term correlation is kept non-squared, to expose the presence or absence of potential negative correlation (per song).

[2] http://mturk.com

[3] http://labrosa.ee.columbia.edu/projects/musicsim/ uspop2002.html

[4] http://developer.echonest.com/downloads

Table 1. Results for SVR with linear kernel (SVR-L), RBF kernel (SVR-RBF) and CCRF

Experiment	Arousal				Valence			
	RMS long	RMS short	Corr long	Corr short	RMS long	RMS short	Corr long	Corr short
SVR-L	0.1965	0.1798	0.6343	0.0116	0.2223	0.1891	0.1733	0.0358
SVR-L relative	0.169	0.1454	0.728	0.0131	0.22	0.1878	0.1601	0.0261
SVR-RBF	0.1942	0.17775	0.6451	0.0111	0.2165	0.1857	0.211	0.0073
SVR-RBF relative	**0.167**	**0.1426**	**0.735**	0.0465	**0.209**	**0.1703**	**0.2965**	0.035
CCRF	0.204	0.176	0.7215	0.0493	0.223	0.1826	0.247	**0.0903**
CCRF relative	0.179	0.1532	0.7176	**0.0707**	0.216	0.1756	0.257	0.0485

We decided to evaluate the long term measures because those tend to be the ones that get reported in the field. The short term measures, on the other hand, are more relevant to the problem of emotion recognition in a song (or any other sequence of frames)—what we should be mainly interested in is how well emotion is predicted in each song, rather than over all of the frames. Long term correlation, on the other hand, shows the overall performance, while potentially hiding bad performance on individual songs.

4.1. Relative representation over the standard approach

The first thing that we tested was whether adding the average of a feature over a song to the feature vector has any effect on the performance of SVR models. The main reason for doing that was to separate the effect of changing the feature representation from absolute to relative and adding the average of a feature to the feature vector. The experiments showed no difference at all in either of the 4 evaluation measures that we used. This gave us a strong confirmation that whatever is the effect that we get, it must come from the different feature representation, and we therefore chose not to use this intermediate representation (with the added average) in any of our further experiments. As the results are identical to those based on the basic representation, we chose not to include them.

Relative representation, on the other hand, does provide a substantial improvement over the performance of SVR-based models, as depicted by the first two rows of the table 1. There is a consistent decrease (14-20%) in RMS for both SVR models in both short and long term measurements, as there is a consistent increase in correlation, again for both the short and long term measurements. The improvement in results for the valence axis is less uniform. We see basically no change in the simple linear SVR model, and only a small decrease in RMS with the RBF kernel, but correlation is still improved substantially both for the short and long term measurements.

4.2. CCRF results

The main effect that CCRF has on the performance of the SVR models, on which it is based, is the increase in correlation. There is a consistent increase in the short term correlation in all of the experiments we ran, and a substantial increase in long term correlation when only the basic feature representation is used. With the relative representation, we see an interesting trend in the results—there is a noticeable decrease in the long term correlation, but the short term correlation is potentially improved.

The increase in short term correlation is not a surprising result—the main idea behind using CCRF is that it tries to exploit temporal dependencies and relationships between different frames, while short term correlation is focusing on the per-song performance of the algorithms.

5. DISCUSSION

The dynamic nature of music and the increasing popularity of the dimensional representation of emotion exposes the need for machine learning techniques that can exploit the temporal relationships present in songs. In addition to that, we still need better models and more suitable features to improve the performance of the algorithms used for emotion prediction in general, and valence prediction in particular.

In this paper we propose a solution to ameliorate each of the problems. Both the adaptation of CCRF and our proposed relative representation offer substantial improvement in performance for all the evaluation metrics used. Interestingly, when used on their own, both techniques have a similar effect on the results, while their combination results in worse RMS and long term correlation. It is probably because both the relative representation and CCRF provide a way of smoothing the predictions, while CCRF focuses more on the temporal aspect and therefore has more of an effect on the per-song measurement. This raises a question that we cannot yet answer. Is short-term correlation more important than the other three metrics or should we focus on a technique that improves the largest number of metrics?

As a comparison with the work done on the same dataset [11], we have calculated the average Euclidean distance between the labels and our predictions (as a percentage of AV space). The results are similar to those achieved by Schmidt *et al.*[11]—ranging between 0.117 (for SVR with rbf kernel and relative feature representation and 0.136 (for CCRF with SVR with the standard feature representation), as compared to 0.160-0.169 achieved by Schmidt *et al.*.

Either way, there is clear evidence that temporally-aware machine learning techniques and more problem-appropriate feature vectors are able to improve the results, although more work needs to be done to find out how the two could be combined. In addition to that, CCRF can easily be used for multimodal emotion recognition making this model an especially attractive option for future work.

6. REFERENCES

[1] BPI, "Digital Music Nation," Tech. Rep., 2013.

[2] D. Bainbridge, S. J. Cunningham, and J. S. Downie, "How People Describe Their Music Information Needs : A Grounded Theory Analysis Of Music Queries," in *Proc. of ISMIR*, 2003, pp. 221–222.

[3] E. Schubert, S. Ferguson, N. Farrar, D. Taylor, and G. E. Mcpherson, "Continuous Response to Music using Discrete Emotion Faces," in *Proc. of CMMR*, 2012.

[4] D. Liu, "Automatic mood detection and tracking of music audio signals," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, pp. 5–18, 2006.

[5] M. D. Korhonen, D. A. Clausi, and M. E. Jernigan, "Modeling emotional content of music using system identification," *IEEE transactions on systems man and cybernetics Part B Cybernetics*, vol. 36, no. 3, 2006.

[6] R. Panda and R. P. Paiva, "Using Support Vector Machines for Automatic Mood Tracking in Audio Music," in *130th Audio Engineering Society Convention*, 2011.

[7] E. M. Schmidt and Y. E. Kim, "Prediction of time-varying musical mood distributions from audio," in *Proc. of ISMIR*, 2010, pp. 465–470.

[8] E. M. Schmidt, D. Turnbull, and Y. E. Kim, "Feature selection for content-based, time-varying musical emotion regression," in *Proc. of ISMIR*. ACM, 2010.

[9] E. Schubert, "Modeling Perceived Emotion With Continuous Musical Features," *Music Perception*, vol. 21, no. 4, pp. 561–585, 2004.

[10] V. R. Carvalho and C.-y. Chao, "Sentiment Retrieval in Popular Music Based on Sequential Learning," *Proc. ACM SIGIR*, 2005.

[11] E. M. Schmidt and Y. E. Kim, "Prediction of Time-Varying Musical Mood Distributions Using Kalman Filtering," *9th ICMLA*, pp. 655–660, 2010.

[12] ——, "Modeling musical emotion dynamics with Conditional Random Fields," *Proc. of ISMIR*, 2011.

[13] M. A. Nicolaou, H. Gunes, and M. Pantic, "Output-associative RVM regression for dimensional and continuous emotion prediction," *Face and Gesture*, 2012.

[14] ——, "Continuous Prediction of Spontaneous Affect from Multiple Cues and Modalities in Valence-Arousal Space," *IEEE Transactions on Affective Computing*, vol. 2, pp. 92–105, 2011.

[15] M. Wöllmer, F. Eyben, S. Reiter, B. Schuller, C. Cox, E. Douglas-Cowie, and R. Cowie, "Abandoning emotion classes - towards continuous emotion recognition with modelling of long-range dependencies." in *INTERSPEECH*. ISCA, 2008.

[16] S. Koelsch, W. A. Siebel, and T. Fritz, "Chapter 12, Functional neuroimaging," in *Handbook of music and emotion theory research application*, P. N. Juslin and J. A. Sloboda, Eds. OUP, 2010, pp. 313–346.

[17] D. J. Hargreaves and A. C. North, "Experimental aesthetics and liking for music," in *Handbook of music and emotions theory research applications*, P. N. Juslin and J. A. Sloboda, Eds. OUP, 2010, ch. 19, pp. 515–547.

[18] H. Gunes and M. Pantic, "Automatic, Dimensional and Continuous Emotion Recognition," *Int'l Journal of Synthetic Emotion*, vol. 1, no. 1, pp. 68–99, 2010.

[19] T. Qin, T.-y. Liu, X.-d. Zhang, D.-s. Wang, and H. Li, "Global Ranking Using Continuous Conditional Random Fields," in *NIPS*, 2008.

[20] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: probabilistic models for segmenting and labelling sequence data," in *ICML*, 2001.

[21] T. Baltrušaitis, N. Banda, and P. Robinson, "Dimensional Affect Recognition using Continuous Conditional Random Fields," in *IEEE FG*, 2013.

[22] C. Sutton and A. Mccallum, *Introduction to Conditional Random Fields for Relational Learning*. MIT Press, 2006.

[23] V. Radosavljevic, S. Vucetic, and Z. Obradovic, "Continuous Conditional Random Fields for Regression in Remote Sensing," in *ECAI*, 2010, pp. 809–814.

[24] J. A. Speck, E. M. Schmidt, B. G. Morton, and Y. E. Kim, "A comparative study of collaborative vs. traditional music mood annotation," *Proc. of ISMIR*, 2011.

[25] Y. E. Kim, D. S. Williamson, and S. Pilli, "Towards quantifying the album effect in artist identification," in *Proc. of ISMIR*, 2006, pp. 393–394.