Cross-Cultural Music Emotion Recognition by Adversarial Discriminative Domain Adaptation

Yi-Wei Chen
Graduate Institue of Communication Engineering
National Taiwan University
Taipei, Taiwan
r02942096@ntu.edu.tw

Yi-Hsuan Yang
Research Center for IT Innovation
Academia Sinica
Taipei, Taiwan
yang@citi.sinica.edu.tw

Homer H. Chen
Graduate Institue of Communication Engineering
National Taiwan University
Taipei, Taiwan
homer@ntu.edu.tw

***Abstract*—Annotation of the perceived emotion of a music piece is required for an automatic music emotion recognition system. Most music emotion datasets are developed for Western pop songs. The problem is that a music emotion recognizer trained on such datasets may not work well for non-Western pop songs due to the differences in acoustic characteristics and emotion perception that are inherent to cultural background. The problem was also found in cross-cultural and cross-dataset studies; however, little has been done to learn how to adapt a model pre-trained on a *source* music genre to a *target* music genre of interest. In this paper, we propose to address the problem by an unsupervised adversarial domain adaptation method. It employs neural network models to make the target music indistinguishable from the source music in a learned feature representation space. Because emotion perception is multifaceted, three types of input feature representations related to timbre, pitch, and rhythm are considered for performance evaluation. The results show that the proposed method effectively improves the prediction of the valence of Chinese pop songs from a model trained for Western pop songs.**

Keywords—Cross-cultural, music emotion, adversarial discriminative domain adaptation.

I. Introduction

With the popularity of on-line music services, a large amount of music pieces created from different corners of the world can be accessed by global audiences. Automatic music emotion recognition (MER) techniques have been developed [1] to facilitate such global music information retrieval by exploiting the appealing feature of music listening that music evokes human mood or emotion.

However, existing MER datasets are mostly created for Western pop songs [2]. Using an MER model trained on such songs to predict the emotion of non-Western music pieces does not yield the best performance [3] due to the cultural differences in acoustic characteristics and emotion perception. Although some efforts have been made to enrich non-Western datasets, the size of such datasets is much smaller than that of Western datasets [2]. To deal with the deficiency, cross-dataset transferring or generalization seems a feasible approach.

Hu and Yang [4] studied the cross-cultural and cross-dataset generalizability of MER models trained on different music features for predicting valence and arousal values, the two principal dimensions of emotions [1]. They found that arousal can be better predicted across datasets than valence.

The issue that a machine learning model pre-trained on a source dataset may not perform well for a target dataset is not specific to MER. It is known as a *domain shift* issue resulting from the data distribution biases between the source dataset and the target dataset [8]. There are two typical machine learning solutions. The first solution fine-tunes the model using the target dataset. However, as the size of the target dataset is usually small, the model may easily overfit. The second solution, referred to as *domain adaptation*, learns a cross-domain invariant feature representation by minimizing the discrepancy between the target data and source data [9]–[11]. It is usually achieved by unsupervised learning. Neither solution has been employed in prior work for MER, to our best knowledge.

In this paper, we study whether and how an unsupervised adversarial domain adaptation method can improve cross-cultural MER. Moreover, as the perception of music emotion is multifaceted, three types of acoustic features related to timbre, pitch, and rhythm are considered for the investigation of cross-cultural generalizability. We conduct experiments on AMG1608 (a Western pop music dataset) and CH818 (a Chinese pop music dataset) used in a similar study [4].

In what follows, we first discuss related work on MER and domain adaptation. Then, we describe the proposed adversarial discriminative domain adaptation method, our network architecture, and experimental settings. Finally, we discuss the experimental results and draw some concluding remarks.

II. Related Work

A. Music emotion recognition

Music emotion is often represented using either the categorical model or the dimensional model developed in music psychology. The categorical model uses a set of discrete mood labels, such as sad and happy, to describe music emotion. Each song is assigned at least one label. The dimensional model represents music emotion in a low-dimensional space, such as valence and arousal [1], by continuous values. The class of MER techniques for predicting the emotion categories of music pieces is referred to as music emotion classification, and the class of MER techniques for predicting the numerical emotion values of music pieces is referred to as music emotion regression. Both classes of techniques have been adopted in many studies, using music of the same genre or cultural background for training and testing [1].

While previous studies focused on training MER models with Western music datasets, some recent work started to investigate whether such models can be directly applied to non-Western music [4], [7], [8]. For example, Hu and Yang [7] explored six music related features for music emotion classification of English and Chinese songs and found that arousal prediction works generally well across datasets, but valence prediction is culture-dependent. Similarly, the study reported by Eerola [8] shows that arousal prediction is generalizable across different musical genres, whereas valence prediction is not.

As acoustic features account for different music characteristics, Hu and Yang [4] further investigated the generalizability of different feature sets for music emotion regression. They found that features related to loudness and timbre have better generalizability for both valence and arousal, but rhythm-related features are only effective for valence and pitch-related features are only effective for arousal. However, transferring useful information from Western music to non-Western music is not explored in these studies.

B. Domain adaptation

Recent domain adaptation methods can be categorized into two approaches. The first approach aims to reweight a model pre-trained on the source domain to make the learned feature representation general enough for the target domain. The learning is performed by minimizing a domain distance metric, such as maximum mean discrepancy [9] or correlation distance [10]. Alternatively, one can also simultaneously train a common representation for classification and reconstruction [11].

Instead of using pre-defined distance metrics, the second approach trains a model to measure the discrepancy between source and target domains, in a data-driven way. Adversarial adaptation methods belong to this approach and have gained popularity recently, after the success of generative adversarial network (GAN) [12]. The goal of a GAN is to estimate a generative model via an adversarial process that simultaneously trains two models: a *generative model G* that tries to generate artificial data with distribution similar to the training data, and a *discriminative model D* that aims to distinguish (through binary classification) between the real data and the data created by *G*. The process is adversarial, because the objective of *D* is to maximize the classification accuracy, whereas the objective of *G* is to minimize the classification accuracy. *D* and *G* are trained iteratively, in a hope that by the end of the process the output of *G* looks similar to the real data. In the same vein, adversarial domain adaptation aims to train a generative model *G* that transforms data from the target domain in such a way that makes *D*, which is a *domain classifier*, believe that the output of *G* are data from the source domain.

Among various adversarial adaptation methods, we choose the adversarial discriminative domain adaptation (ADDA) method [5] in this work, because it has been proven successful for various transfer learning tasks. There are several extensions [13], but ADDA is one of the earliest methods of its kind and does not require a generative model.

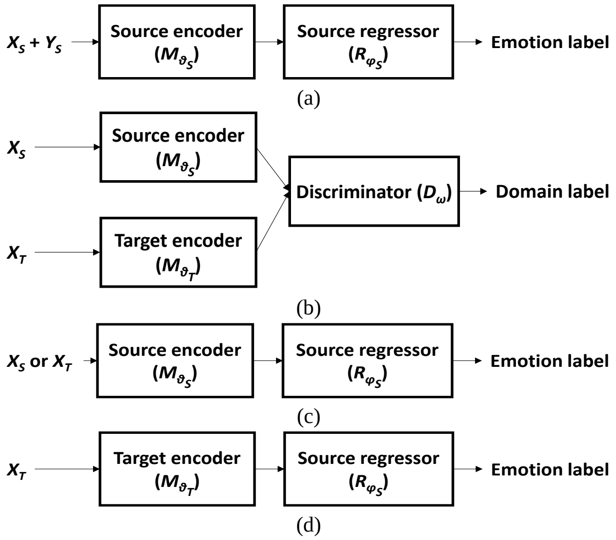

$X_S + Y_S$
Source encoder (M_{ϑ_S})
Source regressor (R_{φ_S})
Emotion label
(a)
X_S
Source encoder (M_{ϑ_S})
Discriminator (D_ω)
Domain label
X_T
Target encoder (M_{ϑ_T})
(b)
X_S or X_T
Source encoder (M_{ϑ_S})
Source regressor (R_{φ_S})
Emotion label
(c)
X_T
Target encoder (M_{ϑ_T})
Source regressor (R_{φ_S})
Emotion label
(d)

Fig. 1. System flow, where X_S are the source data, Y_S are the source emotion labels, and X_T are the target data. (a) The system flow of pre-training an MER model. (b) The system flow of adversarial discriminative domain adaptation. (c) The system flow of testing a dataset without adaptation. (d) The system flow of testing a dataset with adaptation.

III. Methodology

In this section, we discuss how to apply ADDA to cross-cultural MER. We assume that we are given the source data X_S (i.e. input audio features), the source labels Y_S (i.e. emotion labels), and the target data X_T, but not the target labels Y_T. The training process of the proposed method is illustrated in Figs. 1(a) and 1(b) and the test process in Fig. 1(d).

A. Pre-training

As shown in Fig. 1(a), the training process starts with a pre-training phase, using data from the source domain only. Given source data X_S and source labels Y_S, we train a deep neural network for emotion prediction. The task of the first few layers (which can be convolutional layers) of the deep neural network perform feature extraction. It projects the input feature representation X_S into a learned feature space. The task of the last few layers (which can be fully connected layers) predict the emotion values based on the learned features. Therefore, we call the first few layers a source encoder and denote it by M_{θ_S}, and the last few layers a source regressor and denote it by R_{φ_S}. The source encoder and the source regressor are trained jointly by minimizing the mean squared error L_r between Y_S and the predicted emotion values,

$$L_r = \frac{1}{N}\sum_{n=1}^{N}\left(y_S{}^{(n)} - R_{\varphi_S}\left(M_{\theta_S}\left(x_S{}^{(n)}\right)\right)\right)^2, \qquad (1)$$

where N denotes the batch size of source data and $\left\{x_S{}^{(n)}, y_S{}^{(n)}\right\}_{n=1}^{N}$ denotes a batch of $\{X_S, Y_S\}$.

As shown in Fig. 1(c), we can feed target data X_T as the input to the source encoder and the source regressor for emotion prediction. But, due to the domain shift issue, the source regressor may not perform well for the target data.

B. *Adversarial discriminative domain adaptation*

As shown in Fig. 1(d), ADDA attempts to address the domain shift issue by learning a *target encoder* M_{θ_T} for the target data. It is assumed that the output $M_{\theta_T}(X_T)$ of the target encoder would have similar distribution as the output $M_{\theta_S}(X_S)$ of the source encoder. If this is achieved, we consider that the domain shift issue is mitigated and that we can use the source regressor to predict the emotion for the target data without the need of training a target regressor.

The key of ADDA is to learn the target encoder. This is achieved by using a *discriminator* D_ω, which takes either the source feature representation $M_{\theta_S}(X_S)$ or the target feature representation $M_{\theta_T}(X_T)$ as input and decides whether the input is from the source domain or the target domain, as shown in Fig. 1(b). In other words, D_ω is a binary domain classifier. If the accuracy of D_ω is low, we consider $M_{\theta_S}(X_S)$ and $M_{\theta_T}(X_T)$ indistinguishable.

The ADDA method alternately trains the target encoder and the discriminator in two steps. First, the source feature representations with a *source domain label* (say, –1; note that domain labels are not emotion labels) and target feature representations with a *target domain label* (say, +1) are taken as the input to the discriminator, and weights of the discriminator are updated to *minimize* a discriminator loss L_d that aims to promote the accuracy of domain classification. In our approach, the Wasserstein metric [14] is chosen as the loss function to avoid adversarial training from gradient vanishing. Accordingly, the discriminator loss L_d is described by

$$L_d = \frac{1}{N}\sum_{n=1}^{N} D_\omega\left(M_{\theta_T}\left({x_T}^{(n)}\right)\right) - D_\omega\left(M_{\theta_S}\left({x_S}^{(n)}\right)\right), \tag{2}$$

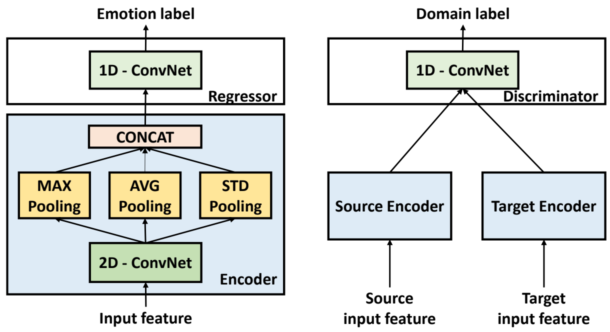
where, as defined in (1), N denotes the batch size of source data and target data, $\left\{{x_S}^{(n)}\right\}_{n=1}^{N}$ denotes a batch of X_S, and, similarly, $\left\{{x_T}^{(n)}\right\}_{n=1}^{N}$ denotes a batch of X_T.

Second, the target feature representations with a flipped domain label (e.g. source domain label becomes +1 and target domain label becomes –1) are taken as input to the discriminator, and weights of the target encoder are updated to *maximize* the discriminator loss. Note that weights of the discriminator are fixed in this step to keep the classification ability of the discriminator. In this way, the target encoder can be trained to fool the discriminator. As the learning objective of the target encoder is at odds with the learning objective of the discriminator, we consider the loss function of the target encoder as *adversarial loss* and denote it by L_a,

$$L_a = -\frac{1}{N}\sum_{n=1}^{N} D_\omega\left(M_{\theta_T}\left({x_T}^{(n)}\right)\right) + D_\omega\left(M_{\theta_S}\left({x_S}^{(n)}\right)\right). \tag{3}$$

Note that the gradient of the second term with respect to θ_T on the right hand side of (3) becomes zero. Therefore, only the target representations in the first term are input to the discriminator in this step.

We repeat the above two steps until the target encoder model is converged. Besides, as Wasserstein loss is applied under a *K*-Lipschitz constraint, weights of the discriminator are clipped into a compact space with absolute supremum C (so ω ranges from $–C$ to C) [14]. To take advantage of the pre-trained source encoder, we use the source encoder as the initial target encoder.



(a) (b)

Fig. 2. Two different network architectures used in our experiments. (a) The network architecture of pre-training MER model. (b) The network architecture of adversarial discriminative domain adaptation.

IV. Network Architecture

The network architecture for pre-training and adaptation is shown in Fig. 2. In this section, we first describe the details of the source encoder, the source regressor, and the discriminator. Then, we describe a simple feature fusion method to improve emotion prediction.

Each song is clipped into 29 seconds to fit the smallest song size and resampled to 22,050 Hz. The input data are three types of acoustic features extracted from these song clips. We use a different 2D convolutional neural network (2D-ConvNet) to encode each type of acoustic feature. The first dimension of the filter in the first convolutional layer is equal to the number of frequency bins of the input feature, because different frequency bins may carry different information of music emotion. Also, we apply three types of pooling (max pooling, average pooling, and standard deviation pooling) to aggregate the output feature maps of these 2D-ConvNets. Because 2D-ConvNets for the three feature inputs use the same number (128) of filters for each layer, the dimensions (128×3) of the concatenated pooling outputs are the same. The concatenation of the pooling outputs is the input to the source regressor and the discriminator.

A. *Log-mel-spectrogram encoder*

We compute the log-mel-spectrogram to extract timbre-related features of the songs, as is the case in many previous works. The spectrogram is first computed with a Hanning window of 1024 samples and a 512-sample stride size and then transformed into a 96-bin log-mel-spectrogram. As a result, the dimensions of the log-mel-spectrogram are 96×1249.

The 2D-ConvNet of the log-mel-spectrogram encoder consists of five convolutional layers. The dimensions of the filters are 96×4, 1×4, 1×3, 1×3, and 1×2, and the filter stride sizes are 1×3, 1×2, 1×3, 1×3, and 1×2 for the five layers. Each convolutional layer is followed by a batch normalization and an ELU activation function.

B. *Pitch salience representation encoder*

We apply the pre-trained deep convolutional network proposed by Bitter *et al.* [15] to extract the pitch salience representation. The goal is to learn the perceived spectral amplitude over time of polyphonic music. Specifically, the harmonic contents are emphasized and the un-pitched or noise contents are de-emphasized to generate the pitch salience representation. Since harmonic summation is usually used to extract pitch content, the network takes harmony-related features extracted by the harmonic constant-Q transform (HCQT) as input. The HCQT generates a time-frequency feature map for each harmonic. The network output has the same size as any harmonic feature map (time-frequency representation).

The frequency dimension of HCQT is partitioned into 360 bins (60 bins per octave for 6 octaves), and the HCQT is computed for 6 harmonic bins using a 512-sample stride size. The resulting 6×360×1249 HCQT feature map is input to the network to generate a 360×1249 pitch salience representation.

Because log-mel-spectrogram and pitch salience representation have the same length in time, we simply change the first dimension of the first filter from 96 to 360 for the 2D-ConvNet and use the same setting for the other filters.

C. Autocorrelation-based tempogram encoder

We compute the autocorrelation-based tempogram through the tempogram toolbox [16] to extract rhythm-related features. Inspired by chromagram, the toolbox applies the concept of tempogram, which is a time-tempo representation for a given time-dependent signal. We adopt an autocorrelation based method with a 0.2-second stride size to extract a 571-bin tempogram, and the resulting dimensions are 571×142, where the first dimension represents the tempo and the second one represents time.

Because the resulting tempogram feature is relatively small, the 2D-ConvNet of the autocorrelation-based tempogram encoder consists of only three convolutional layers. The dimensions of the filters are 571×4, 1×3, and 1×3, and the filter stride sizes are 1×3, 1×2, and 1×2.

D. Regressor and discriminator

As described earlier, the pooling outputs of the source encoder are concatenated into a 128×3 source representation so that the subsequent network can assign individual weights to the three pooling outputs. The resulting representation is input to a regressor consisting of a three-layer 1D convolutional neural network (1D-ConvNet) to recognize the emotion values. The 1D ConvNet of the regressor has 64, 128, and 256 filters for the three layers, the corresponding dimension of filters are 8, 4, and 2, and the corresponding filter stride sizes are 4, 2, and 1. The 1D ConvNet is activated by an ELU at each layer. The output feature maps are flattened to 1D and activated by a tanh neuron to predict emotion values ranging from –1 to 1. The same 1D-ConvNet is used for the discriminator except that the last tanh activation neuron is replaced by a linear activation neuron for computing the Wasserstein loss.

E. Fusion

Because each predicted emotion label is a single value, our fusion method simply takes the average of the predicted emotion values for MER models that use different input features.

V. EXPERIMENTS SETTING

For evaluating the pre-trained MER models, performances were averaged across 10-fold cross validation. As only one dataset was used for training and testing, we call the experiment *within-dataset experiment*. To test if our adaptation method can reduce the domain shift effect, we compared performances between the pre-trained models and the adapted models by averaging performances across 10 segmentations of the target dataset. As datasets used for training and testing are different, we call the experiment *cross-dataset experiment*.

A. Datasets

We chose AMG1608 as our source English dataset and CH818 as our target Chinese dataset. The AMG1608 dataset consists of 1,608 Western song clips, 30 seconds long each [17]. Each clip was obtained from the popular music stream service 7digital and annotated with valence and arousal values ranging from –1 to 1.

The CH818 dataset contains 818 clips of Chinese pop songs [7]. Specifically, the most emotional 30-second segment of each song was extracted through a pre-trained regression model and used as stimuli. Each clip was annotated with valence and arousal values ranging from –10 to 10. We normalized the emotion annotations to [–1, 1].

B. Training parameters

The regression model was trained by using the Adam optimizer with α = 0.001, β_1 = 0.9, and β_2 = 0.999, and 300 epochs. The ADDA was trained by using the RMSProp optimizer with α = 0.001, 2000 epochs, clipping value 0.01, and five iterations of discriminator. The batch size is set to 16 for both trainings.

C. Baseline

The method proposed by Hu and Yang [4] that explored cross-dataset generalizability was adopted as the baseline. Three types of single feature input (including features related to timbre, pitch, and rhythm) and multiple feature inputs were used for comparison. For timbre-related feature, our method using log-mel-spectrogram was compared with the baseline method using dissonance. For pitch-related feature, our method using pitch salience representation feature was compared with the baseline method using log-chromagram. For rhythm-related feature, our method using autocorrelation-based tempogram was compared with the baseline method using autocorrelation-based cyclic tempogram. For multiple-feature inputs, our fusion method using combinations of different feature-predictions was compared with the baseline method using the combined feature set.

Two metrics were used for the regression performance evaluation. The first metric R^2 is the square of correlations between predicted values and ground truth values. The second metric *RMSE* is the root mean squared error between predicted values and ground truth values. Note that performances

Table 1. The R^2 performance of our pre-trained model using different features evaluated in the within-dataset experiment (tested on AMG1608).

	Timbre	Pitch	Rhythm	Timbre + Pitch	Timbre + Rhythm	Rhythm + Pitch	Timbre + Pitch + Rhythm
Valence	**0.24**	0.22	0.05	**0.32**	0.22	0.21	0.31
Arousal	**0.82**	0.68	0.26	**0.82**	0.73	0.65	0.79

Table 2. Comparison of the R^2 performance of our model with or without adaptation evaluated in the cross-dataset experiment (tested on CH818).

	Adaptation	Timbre	Pitch	Rhythm	Timbre + Pitch	Timbre + Rhythm	Rhythm + Pitch	Timbre + Pitch + Rhythm
Valence	-	0.03	0.08	0.04	0.08	0.05	0.08	0.09
	V	**0.21**	0.18	0.06	0.22	0.22	0.17	**0.23**
Arousal	-	0.72	0.69	0.39	0.74	0.68	0.67	0.74
	V	**0.73**	0.65	0.28	**0.76**	0.65	0.49	0.71

Table 3. Comparison of the *RMSE* performance of our model with or without adaptation evaluated in the cross-dataset experiment (tested on CH818).

	Adaptation	Timbre	Pitch	Rhythm	Timbre + Pitch	Timbre + Rhythm	Rhythm + Pitch	Timbre + Pitch + Rhythm
Valence	-	0.39	0.38	0.40	0.38	0.39	0.38	0.38
	V	**0.12**	0.38	0.38	0.35	0.35	0.36	0.35
Arousal	-	0.44	0.43	0.39	0.43	0.39	0.39	0.40
	V	**0.18**	0.42	0.36	0.40	0.31	0.34	0.33

measured by *RMSE* are shown only for our method because Hu and Yang [4] did not use the metric for evaluation.

VI. Results and discussion

In the experiments, we first evaluate the performance of our pre-trained MER model using different features in the within-dataset experiment, where AMG1608 was applied with 10-fold cross validation. Then, we evaluate the effectiveness of adaptation in the cross-dataset experiment, where our model was pre-trained on AMG1608 and tested on CH818. Finally, we make a performance comparison between our model and the model proposed by Hu and Yang [4].

A. *Performance of our pre-trained model*

Table 1 shows the R^2 performance for our pre-trained model based on the timbre, pitch, rhythm, and their combination features. We can observe that the timbre feature is the best single feature for valence prediction (R^2 = 0.24), and the combination of the timbre and pitch features achieves the best performance for valence prediction (R^2 = 0.32). It can be found that combining any feature with the timbre feature can improve the performance. These results are consistent with previous findings [18], [19]: The timbre feature (i.e. log-mel-spectrogram) is effective for many music-related tasks. In our case, combining the log-mel-spectrogram and the pitch salience representation achieves the best performance for valence prediction.

Similar to the valence prediction, the timbre feature is the best single feature for arousal prediction (R^2 = 0.82). The combination of the timbre and pitch features results in the best performance for arousal prediction (R^2 = 0.82), but it does not improve over the single feature. It is worth noting that the worst performance happens to the rhythm feature (R^2 = 0.26), although rhythm seems relevant to arousal in music psychology (e.g. fast songs have high arousal values). The same result was found in previous MER studies [3], [4].

B. *Effectiveness of adaptation*

The R^2 performances of our model with or without adaptation in the cross-dataset experiment are listed in Table 2. We can see that, without adaptation, a large performance drop is resulted. The result shows that the adaptation improves the performance of valence prediction for all features and enables our model to achieve the best performance of arousal prediction for the timbre feature. Among the single features, the timbre feature achieves the best valence prediction performance (R^2 = 0.21). Among the combinations, the combination of all the three features achieves the best valence prediction performance (R^2 = 0.23). For arousal prediction, the timbre feature achieves the best performance (R^2 = 0.73) among the single features, and the combination of the timbre and the pitch features achieves the best performance (R^2 = 0.76) among the combined features. Comparing Tables 1 and 2, we can see that our adapted model indeed improves the pre-trained model for cross-cultural MER performance.

In the case of arousal prediction based on single feature, the performance of the adapted model drops for the pitch and the rhythm features. Perhaps, this is because that arousal is more generalizable across datasets than valence, as reported in the previous studies [7], [8]. This is evident from the experimental result that, for the pitch and the rhythm features, the performance of within-dataset arousal prediction for our pre-trained model is comparable with that of cross-dataset prediction. In other words, there is little domain shift between different datasets when considering the pre-trained models for these two features. As a result, our unsupervised adaptation is unable to improve the pre-trained models further.

However, for the timbre feature, the within-dataset arousal prediction performance of our pre-trained model is significantly better than that of the cross-dataset performance, meaning that we can exploit adaptation to handle the domain shift between different datasets. Indeed, our unsupervised adaptation improves the pre-trained model.

The cross-dataset *RMSE* performances of our model with or without adaptation are listed in Table 3. We can see that our adapted model for the timbre feature improves the pre-trained model a lot (*RMSE* = 0.12 for valence prediction; *RMSE* = 0.18 for arousal prediction), while our adapted model for the pitch and the rhythm features does not get much improvement. As a result, any combination with the pitch and the rhythm features results in little improvement.

Table 4. Comparison of the best R^2 performance of our model and the baseline evaluated in the within-dataset experiment (tested on AMG1608).

	Model	Feature	R^2
Valence	Hu and Yang [4]	Timbre	0.10
	Ours without adaptation	Timbre	**0.24**
	Hu and Yang [4]	Timbre + Loudness + Harmony	0.14
	Ours without adaptation	Timbre + Pitch	**0.32**
Arousal	Hu and Yang [4]	Timbre	0.68
	Ours without adaptation	Timbre	**0.82**
	Hu and Yang [4]	Timbre + Rhythm	0.73
	Ours without adaptation	Timbre + Pitch	**0.82**

Table 5. Comparison of the best R^2 performance of our model and the baseline evaluated in the cross-dataset experiment (tested on CH818).

	Model	Feature	R^2
Valence	Hu and Yang [4]	Rhythm	0.18
	Ours with adaptation	Timbre	**0.21**
	Hu and Yang [4]	Timbre + Loudness + Harmony	0.21
	Ours with adaptation	Timbre + Pitch	**0.23**
Arousal	Hu and Yang [4]	Pitch	0.71
	Ours with adaptation	Timbre	**0.73**
	Hu and Yang [4]	Timbre + Rhythm	0.68
	Ours with adaptation	Timbre + Pitch	**0.76**

C. Comparison of our method and the baseline

Table 4 shows the comparison of the best within-dataset performance for our method and the baseline. Whether for valence prediction or for arousal prediction, our pre-trained model performs better than the baseline for the single features and the combinations. The reason may be that the convolutional neural networks used in our method are able to handle high-dimensional raw features and to learn the representative mid-level features.

Table 5 shows the comparison of the best cross-dataset performance for our method and the baseline. Similar to the within-dataset experiment, whether for valence prediction or for arousal prediction, our adapted model also performs better than the baseline for the single features and the combinations.

Comparing Tables 4 and 5, the timbre feature achieves the best within-dataset performance for our method and the baseline. However, for the best cross-dataset performance, the same feature could be used for our adapted model but could not be used for the baseline. Perhaps, this is because that our adapted model aims to reduce the domain shift but the baseline aims to select the best-performed feature for cross-dataset MER.

VII. Conclusions

This study has explored cross-dataset adaptation of music emotion recognition by adversarial discriminative domain adaptation (ADDA). For cross-dataset experiment, our adapted model does improve for valence prediction but not for arousal prediction, possibly because arousal prediction is more generalizable across datasets. Also, our method performs better than the method proposed by Hu and Yang [4] for both valence prediction and arousal prediction. For future work, we want to experiment on a small number of labeled target data for few-shot learning [20], to analyze what are the musical features that are actually adapted by ADDA, and to experiment with other domain adaptation methods. Moreover, the present method only accounts for the cultural differences in music features, but not for the cultural differences in emotion perception. This is a subject of future work as well.

References

[1] Y.-H. Yang and H. H. Chen, "Machine recognition of music emotion: A review," *ACM Trans. Intelligent Sys. Technology*, vol. 3, no. 3, p. 40, 2012.

[2] X. Hu, and Y.-H. Yang, "The mood of Chinese Pop music: Representation and recognition," *J. Association for Information Science and Technology*, vol. 68, no. 8, pp. 1899–1910, 2017.

[3] K. Kosta, Y. Song, G. Fazekas, and M. B. Sandler, "A study of cultural dependence of perceived mood in Greek music," in *Proc. Int. Soc. Music Information Retrieval*, pp. 1–6, 2013.

[4] X. Hu, and Y.-H. Yang, "Cross-dataset and cross-cultural music mood prediction: A case on Western and Chinese Pop songs," in *IEEE Trans. Affective Comput.*, vol. 8, no. 2, pp. 228–240, 2017.

[5] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. Comput. Vision and Pattern Recognition*, vol. 1, no. 2, p. 4, 2017.

[6] G. Csurka, "Domain adaptation for visual applications: A comprehensive survey," *arXiv preprint arXiv:1702.05374*, 2017.

[7] Y. -H. Yang and X. Hu, "Cross-Cultural music mood classification: A comparison on English and Chinese songs," in *Proc. Int. Soc. Music Information Retrieval*, pp. 19–24, 2012.

[8] T. Eerola, "Are the emotions expressed in music genre-specific? An audio-based evaluation of datasets spanning classical, film, pop and mixed genres," *J. New Music Research*, vol. 40, no. 4, pp. 349–366, 2011.

[9] E. Tzeng *et al.*, "Deep domain confusion: Maximizing for domain invariance," *arXiv preprint arXiv:1412.3474*, 2014.

[10] B. Sun, and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. European Conf. Comput. Vision*, pp. 443–450, 2016.

[11] M. Ghifary *et al.*, "Deep reconstruction-classification networks for unsupervised domain adaptation," in *Proc. European Conf. Comput. Vision*, pp. 597–613, 2016.

[12] I. Goodfellow *et al*., "Generative adversarial nets", in *Advances in Proc. Neural Information Process. Sys.*, pp. 2672–2680, 2014.

[13] J. Hoffman *et al.*, "CyCaDa: Cycle-consistent adversarial domain adaptation," *arXiv preprint arXiv:1711.03213*, 2017.

[14] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[15] R. M. Bittner *et al.*, "Deep salience representations for f0 estimation in polyphonic music," in *Proc. Int. Soc. Music Information Retrieval*, pp. 23‑27, 2017.

[16] P. Grosche, M. Muller, and F. Kurth, "Cyclic tempogram—a midlevel tempo representation for music signals," in *IEEE Trans. Acoustics Speech Signal Process.*, pp. 5522‑5525, 2010.

[17] Y.-A. Chen, Y.-H. Yang, J. C. Wang, and H. Chen, "The AMG1608 dataset for music emotion recognition," in *Proc. IEEE Trans. Acoustics, Speech and Signal Process.*, 2015.

[18] S. Dieleman and B. Schrauwen, "Multiscale approaches to music audio feature learning," in *Proc. Int. Soc. Music Information Retrieval*, pp. 116-121, 2013.

[19] A. Van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Advances in Proc. Neural Information Process. Sys.*, pp. 2643-2651, 2013.

[20] S. Ravi, and H. Larochelle, "Optimization as a model for few-shot learning", in *Proc. Int. Conf. Learning Repres.*, 2017.