Adversarial and Cooperative Correlated Domain Adaptation based Multimodal Emotion Recognition

Jie-Lin Qiu[1], Xiaoshi Chen[2], and Kai Hu[1]*

[1]Shanghai Jiao Tong University, Shanghai, China
[2]Southeast University, Nanjing, China
{sjtu_hukai@sjtu.edu.cn}

Abstract. In this paper, we propose a new model, Adversarial and Cooperative Correlated Domain Adaptation (ACCDA), to make multimodal emotion recognition. Adversarial and Cooperative Correlated Domain Adaptation (ACCDA) is an extension and unity, which unifies adversarial discriminative domain adaptation and cooperative generative domain adaptation with deep canonical correlation analysis to train highly correlated domains of multiple physiological data (EEG and eye movement signals), which make use of their complentarity and relevance. In experiments on two real world datasets, we find that our model can significantly contribute to higher emotion classification accuracy when higher correlation is acquired. Our experiment results indicate that the Adversarial and Cooperative Correlated Domain Adaptation model performs better than the state-of-the-art methods with a mean accuracy of 88.64% for four emotion classification on SEED IV dataset. It also outperforms than the state-of-the-art results on DEAP dataset with a mean accuracy of 86.15% for two dichotomies.

Keywords: Emotion Recognition, EEG, Eye Movement, Domain Adaptation

1 Introduction and Related Work

Multimodal emotion recognition from electroencephalography (EEG) and eye movement features have attracted increasing interest. Integrating this information with fusion technologies is attractive for constructing robust emotion recognition models. The combination of signals from the central nervous system, EEG, and external behaviors, eye movement, has been reported to be a promising approach [1,2,3,4,5,6,7,8,9,21].

Domain adaptation methods attempt to mitigate the harmful effects of domain shift. Recent domain adaptation methods learn deep neural transformations that map both domains into a common feature space. This is generally achieved by optimizing the representation to minimize some measure of domain shift such as maximum mean discrepancy [11,12] or correlation distances [13,14]. An alternative is to reconstruct the target domain from the source representation [15]. Adversarial adaptation methods have become an increasingly popular incarnation of this type of approach which seeks to minimize an approximate domain discrepancy distance through an adversarial objective with respect to a domain discriminator. These methods are closely related to generative adversarial learning [16], which pits two networks against each other a generator

* corresponding author

and a discriminator. The generator is trained to produce images in a way that confuses the discriminator, which in turn tries to distinguish them from real image examples. In domain adaptation, this principle has been employed to ensure that the network cannot distinguish between the distributions of its training and test domain examples [17,18]. However, each algorithm makes different design choices such as whether to use a generator, which loss function to employ, or whether to share weights across domains. For example, [17] share weights and learn a symmetric mapping of both source and target images to the shared feature space, while [18] decouple some layers thus learning a partially asymmetric mapping. Eric Tzeng *et al.* combined discriminative modeling, untied weight sharing, and a GAN loss to form ADDA model and outperformed unsupervised adaptation results [19].

However, since the gradient computation requires back propagation through the generators output, GAN can only model the distribution of continuous variables, making it non-applicable for discrete sequences generation. Researchers then proposed Sequence Generative Adversarial Network (SeqGAN) [18], which uses model-free policy gradient algorithm to optimize the original GAN objective. With SeqGAN, the expected JSD between current and target discrete data distribution is minimized if the training is perfect. SeqGAN shows observable improvements in many tasks. Since then, many variants of SeqGAN have been proposed to improve its performance. However, SeqGAN is not an ideal algorithm for this problem, and current algorithms based on it cannot show stable, reliable and observable improvements that cover all scenarios. So Lu *et al.* proposed CoT for training generative models that measure a tractable density function for target data [20]. For multimodal emotion, Lu *et al.* used both EEG signals and eye movement signals to recognize three types of emotions [21]. Liu *et al.* furthermore used Bimodal Deep AutoEncoder to extract high level representation features [22]. Tang *et al.* adopted the Bimodal Deep Denoising AutoEncoder modal, taking Bimodal-LSTM model into account [23].

In this paper, we combine adversarial and cooperative networks to propose a new domain adaptation framework, named Adversarial and Cooperative Correlated Domain Adaptation (ACCDA), where correlation calculated between different models in high dimension which can take advantage of complementary of multiple models and tends out to achieve remarkable results. Our results demonstrate the complementary of adversarial and cooperative networks, which indicates a new direction for multiple model based tasks.

2 Adversarial and Cooperative Correlated Domain Adaptation

2.1 Background

Domain adaptation

DA is a branch of transfer learning (i.e., transductive learning within the same feature space [24]). The source domain is denoted by $D_s = \{X_s, Y_s\}$, in which $X_s = \{x_{s1}, x_{s2}, ..., x_{sn}\}$ is the input and $Y_s = \{y_{s1}, y_{s2}, ..., y_{sn}\}$ is the corresponding label set. The values of X_s and Y_s are drawn from the joint distribution $P(X_s, Y_s)$. Similarly, the target domain denoted by $D_t = \{X_t, Y_t\}$ corresponds to data and labels drawn from the joint distribution $P(X_t, Y_t)$. In this paper, we consider unsupervised domain

adaptation, which means label information from the target domain is not required. Typically, the marginal distributions of the input data are different between source domain and target domain: $P(X_s) \neq P(X_t)$. This is usually referred to as domain shift and is considered to be the key problem that leads to poor performance when a model is trained and tested on data from different domains. To eliminate the influence of domain shift, feature-based domain adaptation methods try to find a proper transformation function $\phi(\cdot)$ that aligns the data into a new feature space where $P(\phi(X_s)) = P(\phi(X_t))$.

Domain-Adversarial Neural Network

DANN was first proposed in [17], and its properties and applications are then further explored in [19]. The model can be divided into the following three parts: a feature extractor G_f, a label predictor G_y, and a domain classifier G_d. There exists adversarial relationships between the feature extractor and the domain classifier. The feature extractor, as the name implies, extracts new features from input features: $f = G_f(x; \theta_f)$. Here x denotes input feature vector and f denotes the corresponding output feature vector in a new feature space. The outputs are then fed into the label predictor and the domain classifier. The label predictor provides predictions of the corresponding labels: $\hat{y} = G_y(f; \theta_y)$. The domain classifier distinguishes which domain the input is from: $\hat{d} = G_d(f; \theta_d)$. The three parts are updated simultaneously with the objective function:

$$E(\theta_f, \theta_y, \theta_d) = \sum_{i=1}^{N} L_y(\hat{y}_i, y_i) - \lambda \sum_{i=1}^{N} L_d(\hat{d}_i, d_i) \tag{1}$$

where the first term $L_y(\cdot, \cdot)$ is the loss for label prediction, and $L_d(\cdot, \cdot)$ corresponds to the loss for domain classification. The update rule is designed as follows:

$$(\hat{\theta}_f, \hat{\theta}_y) = arg \min_{\theta_f, \theta_y} E(\theta_f, \theta_y, \hat{\theta}_d),\ \hat{\theta}_d = arg \max_{\theta_d} E(\hat{\theta}_f, \hat{\theta}_y, \theta_d) \tag{2}$$

It can be observed that the label predictor and domain classifier are trained so that the corresponding losses are minimized. The feature extractor is trained so that the label prediction loss is minimized while the domain classification loss is maximized. So the feature extractor is trying to extract features that are good for label prediction, but not easy to distinguish which domain the features come from. In this way, the feature extractor is to extract domain invariant features, so the domain shift can be eliminated.

Adversarial Discriminative Domain Adaptation

Similar to DANN, ADDA also can be divided into three parts, except that there are two feature extractors, one for source domain data and another for target domain data [19]. Let G_{f0} and G_{f1} be the corresponding feature extractors for source domain and target domain, respectively. The training procedure is two-stage. In the first stage, G_{f0} and the label predictor G_y are trained with source domain data so that the prediction loss is minimized. After the training, the parameters of G_{f0} and G_y are fixed during the following process. In the second stage, G_{f1} is initialized with the parameters of G_{f0}. Then G_{f1} and G_d are trained adversarially: G_d is trained to discriminate source domain data and target domain data, while G_{f1} is trained to fool G_d. So, after the training, the feature extractor G_{f1} aligns the distribution of the target domain data to that of the source domain data.

Canonical Correlation Analysis

Canonical correlation analysis is an algorithm to learn the non-linear transformation of parameters of two random vectors in order to maximize the correlation between them [25]. Let $(X_1, X_2) \in R^{n1} \times R^{n2}$ denote random vectors with covariances$(\Sigma_{11}, \Sigma_{22})$ and crosscovariance Σ_{12}. CCA finds maximally correlated pairs of linear projections of the two views,$(\omega_1{}'X_1, \omega_2{}'X_2)$:

$$(\omega_1^*, \omega_2^*) = \underset{\omega_1,\omega_2}{\arg\max}\, corr(\omega_1{}'X_1, \omega_2{}'X_2) = \underset{\omega_1,\omega_2}{\arg\max} \frac{\omega_1{}'\Sigma_{12}\omega_2}{\sqrt{\omega_1{}'\Sigma_{11}\omega_1\omega_2{}'\Sigma_{22}\omega_2}} \tag{3}$$

Since the objective is invariant to scaling of w_1 and w_2, we can limit the projections with unit variance:

$$(\omega_1*, \omega_2*) = \underset{\omega_1{}'\Sigma_{11}\omega_1=\omega_2{}'\Sigma_{22}\omega_2=1}{\arg\max} \omega_1{}'\Sigma_{12}\omega_2 \tag{4}$$

When finding multiple pairs of vectors $({\omega_1}^i, {\omega_2}^i)$,subsequent projections are also constrained to be uncorrelated with previous ones, which is ${\omega_1}^i\Sigma_{11}{\omega_1}^j = {\omega_2}^i\Sigma_{22}{\omega_2}^j = 0$ for $i < j$. Assembling the top k projection vectors ${\omega_1}^i$ into the columns of a matrix $A_1 \in R^{n_1 \times k}$, and similarly placing ${\omega_2}^i$ into $A_2 \in R^{n_2 \times k}$, we obtain the following formulation to identify the top $k \leq min(n_1, n_2)$ projections:

$$maximize : tr(A_1{}'\Sigma_{12}A_2),\ subject\ to : A_1{}'\Sigma_{11}A_1 = A_2{}'\Sigma_{22}A_2 = I \tag{5}$$

Cooperative Generative Model

Lu *et al.* proposed Cooperative Training (CoT) for training generative models that measure a tractable density function for target data [20]. CoT coordinately trains a generator G and an auxiliary predictive mediator M. The training target of M is to estimate a mixture density of the learned distribution G and the target distribution P, and that of G is to minimize the Jensen-Shannon divergence estimated through M. CoT achieves independent success without the necessity of pre-training via Maximum Likelihood Estimation or involving high-variance algorithms like REINFORCE. This low-variance algorithm is theoretically proved to be unbiased for both generative and predictive tasks.

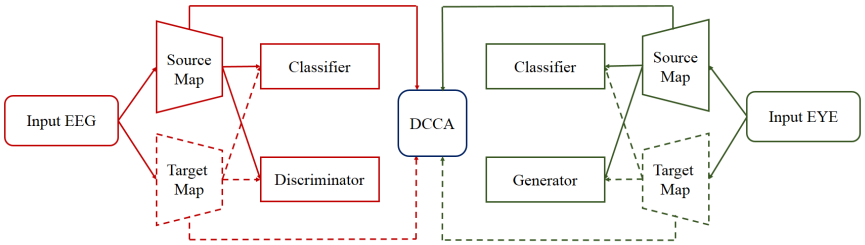

Fig. 1: Adversarial and Cooperative Correlated Domain Adaptation Networks. The left part is ADDA network, the right part is CGDA network, and the DCCA network is in the middle.

2.2 Our Model

The overall architecture of the Adversarial and Cooperative Correlated Domain Adaptation (ACCDA) is shown in Figure 1. There are two views of networks which contain EEG signals and eye movement signals respectively. It consists of three parts: an adversarial discriminative domain adaptation (ADDA) network, an cooperative generative domain adaptation (CGDA) network, and a deep canonical correlation analysis network. The two domain adaptation networks will be trained simultaneously and independently, the DCCA network is applied to extract more highly correlated source and target map of both views. We describe the details of different components in the following paragraphs.

For adversarial discriminative domain adaptation network, we first pre-train a source encoder network using labeled source examples. Next, we perform adversarial adaptation by learning a target encoder network such that a discriminator that sees encoded source and target examples cannot reliably predict their domain label. During testing, target images are mapped with the target encoder to the shared feature space and classified by the source classifier. Source network's pre-trained network parameters will be fixed and transmitted to target network. In unsupervised adaptation, we assume access to source images X_s and labels Y_s drawn from a source domain distribution $p_s(x, y)$, as well as target images X_t drawn from a target distribution $p_t(x, y)$, where there are no label observations. Domain adaptation instead learns a source representation mapping, M_s , along with a source classifier, C_s, and then learns to adapt that model for use in the target domain. We regularize the learning of the source and target mappings, M_s and M_t, so as to minimize the distance between the empirical source and target mapping distributions: $M_s(X_s)$ and $M_t(X_t)$. The source classification model is then trained using the standard supervised loss:

$$\min_{M_s,C} L_{cls}(X_s, Y_t) = \mathbb{E}_{(x_s,y_s)\sim(X_s,Y_s)} - \sum_{k=1}^{K} \mathbb{I}_{[}y = k_s] log C(M_s(x_s)) \tag{6}$$

A domain discriminator D is optimized according to a standard supervised loss:

$$L_{adv_D}(X_s, X_t, M_s, M_t) = -\mathbb{E}_{x_s\sim X_s}[logD(M_s(x_s))] - \mathbb{E}_{x_t\sim X_t}[log(1 - D(M_s(x_s)))] \tag{7}$$

Once the mapping parameterization is determined for the source, target mapping is set so as to minimize the distance between the source and target domains under their respective mappings, while crucially also maintaining a target mapping that is category discriminative. Consider a layered representations where each layer parameters are denoted as, M_s^l or M_t^l, for a given set of equivalent layers, $\{l_1, ..., l_n\}$. Then the space of constraints explored in the literature can be described through layerwise equality constraints as follows:

$$\psi(M_s, M_t) \triangleq \{\psi_{l_i}(M_s^{l_i}, M_t^{l_i})\}_{i\in\{1,...,n\}} \tag{8}$$

where each individual layer can be constrained independently. A very common form of constraint is source and target layerwise equality:

$$\psi_{l_i}(M_s^{l_i}, M_t^{l_i}) = (M_s^{l_i} = M_t^{l_i}) \tag{9}$$

We choose to allow independent source and target mappings by untying the weights. This is a more flexible learning paradigm as it allows more domain specific feature extraction to be learned. However, note that the target domain has no label access, and thus without weight sharing a target model may quickly learn a degenerate solution if we do not take care with proper initialization and training procedures. Therefore, we use the pre-trained source model as an initialization for the target representation space and fix the source model during adversarial training. In doing so, we are effectively learning an asymmetric mapping, in which we modify the target model so as to match the source distribution. This is most similar to the original generative adversarial learning setting, where a generated space is updated until it is indistinguishable with a fixed real space. Therefore, we choose the inverted label GAN loss:

$$L_{adv_M}(X_s, X_t, D) = -\mathbb{E}_{x_t \sim X_t}[logD(M_t(x_t))] \tag{10}$$

As for cooperative generative domain adaptation network, inspired by Lu *et al.*'s work [20], we set the CGDA with similar structure compared with adversarial discriminative domain adaptation network, only replace discriminator with generator. So the domain generator G loss is:

$$L_{gen_G}(X_s, X_t, M_s, M_t) = \mathbb{E}_{s \sim p_{data}}[log(M_\phi(x_s))] + \mathbb{E}_{s \sim G_\theta}[log(M_\phi(x_s))] \tag{11}$$

where M_ϕ is the mediator, a predictive module that measures a mixture distribution of the learned generative distribution G_θ and target latent distribution $P = p_{data}$ as $M_\phi = \frac{1}{2}(P + G_\theta)$.

Table 1: Value sets for hyperparameter tuning.

Type	Value Set
Subspace Dimension	{10,20,40,60,80,100,120}
λ for ADDA and CGDA	$\{2^n \vert n \in \{-10, -9, ..., 10\}\}$
Regulation Parameters	$\{1e^n \vert n \in \{-10, -9, ..., -3\}\}$
Learning Rate for Adam	$\{2^n \times 10^{-4} \vert n \in \{-10, -9, ..., 10\}\}$
Learning Rate for DCCA	$\{10^n \vert n \in \{-5, -4, ..., -1\}\}$
DCCA Layers	$[400 \pm 40, 200 \pm 20, 150 \pm 20,$ $120 \pm 10, 60 \pm 10, 20 \pm 2]$

For deep canonical correlation analysis, we take advantage of complementarity of EEG and eye movement signals and train an DCCA network to extract highly correlated domain of both views. DCCA is proposed by Galen Andrew *et al.*, which is a non-linear version of CCA that uses neural networks as the mapping functions instead of linear transformers [26]. DCCA directly optimizes the correlation between the two views' potential learning representation. Retrieval can be performed by the cosine distance when given the correlated embedding representations of the two views. We regard the source and target domain of two views as input respectively. The layer size of both views are the same, including input layer L_1, hidden layers L_2, and output layer

L_3 with nodes of each layer are fully connected. When training, we first use the deep networks to extract features, then we calculate the correlation at the output layer with canonical correlation analysis. The goal is to jointly learn parameters for both views' W and b, where $W \in R^{c_1 \times n_1}$ is a matrix of weights, $b \in R^{c_1}$ is a vector of biases, and c_1 is the units of each intermediate layer in the network for the first view, such that $corr(f_1(X_1), f_2(X_2))$ is as high as possible, where $f(\cdot)$ is the whole function of each view's network. We define H_1 and H_2 matrices whose columns are the top-level representations produced by the deep models on the two views in layer L_3 with k eigenvalues, and the total correlation of H_1 and H_2 is the sum of the k singular values of the matrix:

$$T = \Sigma_{11}^{-1/2}\Sigma_{12}\Sigma_{22}^{-1/2},\ l_{corr} = corr(H_1, H_2) = ||T||_{tr} = tr(T'T)^{1/2} \tag{12}$$

Weights of nodes update using back propagation. The DCCA parameters W_l^v and b_l^v are trained to use gradient-based optimization to optimize this quantity.

$$\frac{\partial corr(H_1, H_2)}{\partial H_1} = \frac{1}{m-1}(2\nabla_{11}\overline{H}_1 + \nabla_{12}\overline{H}_2) \tag{13}$$

where

$$\nabla_{12} = \hat{\Sigma}_{11}^{-1/2}UV'\hat{\Sigma}_{22}^{-1/2},\ \nabla_{11} = -\frac{1}{2}\hat{\Sigma}_{11}^{-1/2}UDU'\hat{\Sigma}_{22}^{-1/2}. \tag{14}$$

3 Experiment

3.1 Dataset

We evaluate the performance of the approaches on two real-world datasets: the SEED IV[1] dataset, and the DEAP[2] dataset [27]. The SEED IV dataset contains EEG and eye movement signals in total of four emotions [28]. There were 72 film clips in total for four emotions and forty five experiments were taken by participants to assess their emotions when watching the film clips with keywords of emotions and ratings out of ten points for two dimensions: valence and arousal. The valence scale ranges from sad to happy. The arousal scale ranges from calm to excited. The EEG signals were recorded with ESI NeuroScan System at a sampling rate of 1000 Hz with a 62-channel electrode cap. The eye movement signals were recorded with SMI ETG eye tracking glasses. The DEAP dataset contains EEG signals and peripheral physiological signals of 32 participants. Signals were collected while participants were watching one-minute-long emotional music videos. We chose 5 as the threshold to divide the trials into two classes according to the rated levels of arousal and valence. We used 5-fold cross validation to compare with Liu *et al.* [22] and Yin *et al.* [29].

[1] http://bcmi.sjtu.edu.cn/∼ seed/

[2] http://www.eecs.qmul.ac.uk/mmv/datasets/deap/

3.2 Feature Extraction

For SEED IV dataset, we extracted Differential Entropy (DE) features from each EEG signal channel in five frequency bands: δ (1-4 Hz), θ (4-8 Hz), α (8-14Hz), β (14-31 Hz) and γ (31-50 Hz). The size of Hanning window used when extracting EEG features was 4 s. At each time step, there were totally 310 (5 bands $\times$ 62 channels) dimensions for EEG features. As for eye movement data, the features used are shown in Fig 2 and Table 2. There were totally 39 dimensions including both Power Spectral Density (PSD) and DE features of pupil diameters at each time step. Before training the model, the features were normalized to zero mean. One view contains EEG features and the other contains eye movement features.

For DEAP dataset, we extracted DE features from EEG signals in four frequency bands: θ (4-8 Hz), α (8-14 Hz), β (14-31 Hz) and γ (31-50 Hz), since a bandpass frequency filter from 4 - 45 Hz was applied during pre-processing. The size of Hanning windows was 2 s. Then there were totally 128 (4 bands $\times$ 32 channels) dimensions of extracted 32-channel EEG features. As for peripheral physiological signals, six time-domain features were extracted to describe the signals in different perspective, including maximum value, minimum value, mean value, standard deviation, variance and squared sum. So there were totally 48 (6 features $\times$ 8 channels) dimensions of extracted peripheral physiological features.

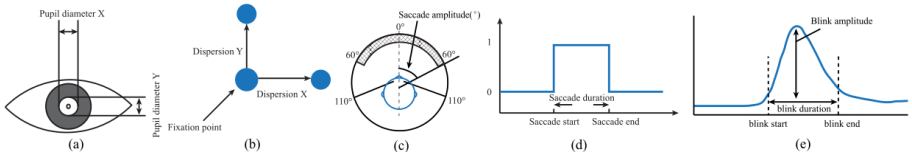

Fig. 2: Illustration of various eye movement parameters: pupil diameter, fixation dispersion, saccade amplitude, saccade duration, and blink.

3.3 Experiment

Our objective now is to perform domain adaptation between different subjects. The leave-one-subject-out cross-validation algorithm is applied, which means, for each domain adaptation method there are a few runs, and for each run the data from one of the subjects are regarded as target domain while the data from other subjects as source domains. Multi-layer perceptrons (MLPs) are used for the feature extractors, the label predictors, and the domain classifiers in the adversarial domain adaptation networks. Adam optimizer [30] was adopted for training of the networks to obtain faster convergence. We performed randomized search of the hyperparameters over some predefined sets of values. For each method, the hyperparameter settings were evaluated with the leave-one-subject-out cross-validation algorithm and the best setting was chosen to generate the final results. For DCCA networks, we use Grid Search to find optimal hyperparameters. The specific predefined value sets for some of the hyperparameters are listed in Table 1. After extracting features by DCCA, we apply SVM for classification.

Table 2: The details of the extracted eye movement features.

Eye movements parameters	Extracted features
Pupil diameter(X and Y)	Mean,standard deviation, DE in four bands (0-0.2Hz, 0.2-0.4Hz, 0.4-0.6Hz, 0.6-1Hz)
Disperson(X and Y)	Mean, standard deviation
Fixation duration (ms)	Mean, standard deviation
Blink duration (ms)	Mean, standard deviation
Saccade	Mean, standard deviation of saccade duration(ms) and saccade amplitude
Event statistics	Blink frequency, fixation frequency, fixation dispersion total, fixation duration maximum, fixation dispersion maximum, saccade frequency, saccade duration average, saccade latency average, saccade amplitude average.

4 Results and Discussion

4.1 Results on Different Datasets

For SEED IV dataset, we regard Zheng *et al.*'s multimodal deep learning results as our baseline [28]. We use different kinds of methods to make comparison with our model. Table 3 demonstrates that BDAE achieved better results than SVM based feature fusion. Compared with CCA based approach and other method, we conclude that ACCDA model which coordinated signals achieved better results. Table 4 shows comparison results of different methods on DEAP dataset. For two dichotomous classification, Liu *et al.*'s multimodal autoencoder model achieved 2% higher than AutoEncoder. Yin *et al.* used an ensemble of deep classifiers, making higher-level abstractions of physiological features [29]. Then Tang *et al.* used Bimodal-LSTM and achieved the state-of-the-art accuracy for two dichotomous classification [23]. As for our ACCDA method, we learned correlation of multiple domain signals and achieved better results than the state-of-the-art method with mean accuracies of 85.86% and 86.45% for arousal and valence classification tasks.

Table 3: Average accuracies (%)and standard deviation of different approaches for four emotion classification on SEED IV dataset

	CCA	SVM	BDAE	ACCDA
Accuracy(%)	49.56	75.88	85.11	**88.64**
Std	19.24	16.14	11.79	**8.53**

Table 4: Comparison of average accuracies (%) of different approaches on DEAP dataset for two dichotomous

	CCA	AutoEncoder	Liu *et al.*	Yin *et al.*	Bimodal-LSTM [23]	ACCDA
Arousal(%)	61.25	74.49	80.5	84.18	83.23	**85.86**
Valence(%)	69.58	75.69	85.2	83.04	83.82	**86.45**

4.2 Discussion

In comparison with previous feature-level fusion and multimodal deep learning method, it is very difficult to relate the original features in one modality to features in other modality and this method usually learns unimodal features [31]. Moreover, the relations across various modalities are deep instead of shallow. In our model, we can learn coordinated representation from high-level signals and make two views of signals become more complementary, which in return improves the classification performance of fusion features. To find out the complementarity of adversarial and cooperative domain adaptation, we verified the performance of multiple networks, which means both view's ADDA networks, both view's CGDA networks to compare with ADDA-CGDA and CGDA-ADDA method.

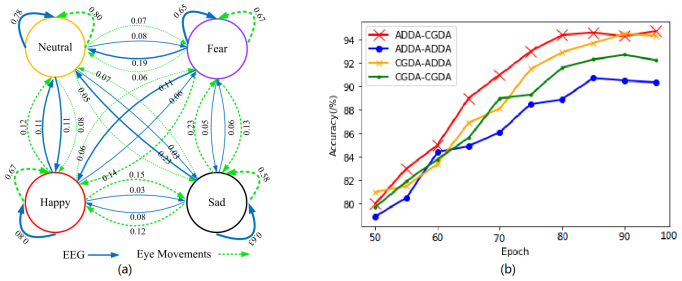

Fig. 3: (a) Confusion graph of EEG and eye movements of SEED IV dataset, which shows their complementary characteristics. The numbers denote the percentage values of samples in the class (arrow tail) classified as the class (arrow head). (b) Accuracies of three methods when epoch number increases on DEAP dataset.

5 Conclusion

In this paper, we proposed a new method, Adversarial and Cooperative Correlated Domain Adaptation (ACCDA), to make multimodal emotion recognition on three real world datasets. The model learns correlation from high-level domains due to the complementarity and relevance of multiple signals. The experimental results have shown that our model contributes to higher classification accuracy of emotion recognition with high correlation.

References

1. Soleymani, M., Pantic, M., Pun, T.: Multimodal emotion recognition in response to videos. IEEE Trans. Affective Computing, 3, 211-223 (2012)
2. DMello, S. K., and Westlund, J. K. 2015. A review and meta-analysis of multimodal affect detection systems. ACM Comput. Surv. 47:43:143:36.
3. Picard, R. W. 1997. Affective computing.
4. Bocharov, A. V.; Knyazev, G. G.; and Savostyanov, A. N. 2017. Depression and implicit emotion processing: An eeg study. Neurophysiologie clinique 47 3:225230.
5. Tzirakis, P.; Trigeorgis, G.; Nicolaou, M. A.; Schuller, B. W.; and Zafeiriou, S. 2017. End-to-end multimodal emotion recognition using deep neural networks. IEEE Journal of Selected Topics in Signal Processing 11:13011309.
6. Hassib, M.; Schneega, S.; Eiglsperger, P.; Henze, N.; Schmidt, A.; and Alt, F. 2017b. Engagemeter: A system for implicit audience engagement sensing using electroencephalography. In CHI.
7. Wang, X.-W.; Nie, D.; and Lu, B.-L. 2014. Emotional state classification from eeg data using machine learning ap- proach. Neurocomputing 129:94106.
8. Hassib, M.; Pfeiffer, M.; Schneega, S.; Rohs, M.; and Alt, F. 2017a. Emotion actuator: Embodied emotional feedback through electroencephalography and electrical muscle stim- ulation. In CHI.
9. Zheng, W.-L.; Dong, B.-N.; and Lu, B.-L. 2014. Multimodal emotion recognition using eeg and eye tracking data. 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society 50405043.
10. Lu,Y.;Zheng,W.-L.;Li,B.;andLu,B.-L. 2015. Combining eye movements and eeg to enhance emotion recognition. In IJCAI.
11. Tzeng, E.; Hoffman, J.; Zhang, N.; Saenko, K.; and Darrell, T. 2014. Deep domain confusion: Maximizing for domain invariance. CoRR abs/1412.3474.
12. Long, M.; Cao, Y.; Wang, J.; and Jordan, M. I. 2015. Learning transferable features with deep adaptation networks. In ICML.
13. Sun, B.; Feng, J.; and Saenko, K. 2016. Return of frustratingly easy domain adaptation. In AAAI.
14. Sun,B.,andSaenko,K. 2016. Deepcoral:Correlationalignment for deep domain adaptation. In ECCV Workshops.
15. Ghifary, M.; Kleijn, W. B.; Zhang, M.; Balduzzi, D.; and Li, W. 2016. Deep reconstruction-classification networks for unsupervised domain adaptation. In ECCV.
16. Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A. C.; and Bengio, Y. 2014. Generative adversarial nets. In NIPS.
17. Ganin, Y., and Lempitsky, V. S. 2015. Unsupervised domain adaptation by backpropagation. In ICML.
18. Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In AAAI.
19. Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 29622971.
20. Lu, S.; Yu, L.; Zhang, W.; and Yu, Y. 2018. Cot: Cooperative training for generative modeling. CoRR abs/1804.03782.
21. Lu,Y.;Zheng,W.-L.;Li,B.;andLu,B.-L. 2015. Combining eye movements and eeg to enhance emotion recognition. In IJCAI.
22. Liu, W.; Zheng, W.-L.; and Lu, B.-L. 2016. Emotion recognition using multimodal deep learning. In ICONIP.

23. Tang, H.; Liu, W.; Zheng, W.-L.; and Lu, B.-L. 2017. Multimodal emotion recognition using deep neural networks. In ICONIP.
24. Qiao, R.; Qing, C.; Zhang, T.; Xing, X.; and Xu, X. 2017. A novel deep-learning based framework for multi-subject emotion recognition. 2017 4th International Conference on Information,CyberneticsandComputationalSocialSystems (ICCSS) 181185.
25. Hotelling, H. 1936. Relations between two sets of variates. Biometrikae.
26. Andrew, G.; Arora, R.; Bilmes, J. A.; and Livescu, K. 2013. Deep canonical correlation analysis. In ICML.
27. Koelstra, S.; Mhl, C.; Soleymani, M.; Lee, J.-S.; Yazdani, A.; Ebrahimi, T.; Pun, T.; Nijholt, A.; and Patras, I. 2012. Deap: A database for emotion analysis ;using physiological signals. IEEE Transactions on Affective Computing 3:18 31.
28. Zheng, W.-L.; Liu, W.; Lu, Y.; liang Lu, B.; and Cichock- i, A. 2018. Emotionmeter: A multimodal framework for recognizing human emotions. IEEE transactions on cyber- netics.
29. Yin, Z.; Zhao, M.; Wang, Y.; Yang, J.; and Zhang, J. 2017. Recognition of emotions using multimodal physiological signals and an ensemble deep learning model. Computer methods and programs in biomedicine 140:93110.
30. Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. CoRR abs/1412.6980.
31. Ngiam, J.; Khosla, A.; Kim, M.; Nam, J.; Lee, H.; and Ng, A. Y. 2011. Multimodal deep learning. In ICML.