Facial Emotion Distribution Learning by Exploiting Low-Rank Label Correlations Locally

Xiuyi Jia[1,2], Xiang Zheng[1], Weiwei Li[3], Changqing Zhang[4], Zechao Li[1*]

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China
[2]State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
[3]College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing, China
[4]College of Intelligence and Computing, Tianjin University, Tianjin, China

Abstract

Emotion recognition from facial expressions is an interesting and challenging problem and has attracted much attention in recent years. Substantial previous research has only been able to address the ambiguity of "what describes the expression", which assumes that each facial expression is associated with one or more predefined affective labels while ignoring the fact that multiple emotions always have different intensities in a single picture. Therefore, to depict facial expressions more accurately, this paper adopts a label distribution learning approach for emotion recognition that can address the ambiguity of "how to describe the expression" and proposes an emotion distribution learning method that exploits label correlations locally. Moreover, a local low-rank structure is employed to capture the local label correlations implicitly. Experiments on benchmark facial expression datasets demonstrate that our method can better address the emotion distribution recognition problem than state-of-the-art methods.

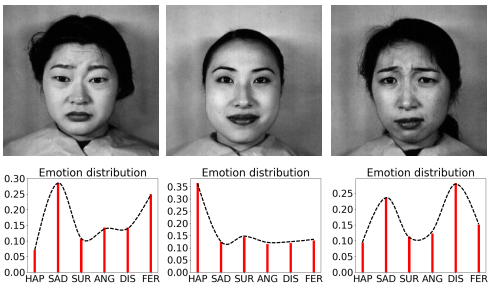
1. Introduction

As one of the most natural, powerful and immediate means for human beings to express their emotions and intentions, facial expression recognition techniques have already been adopted in numerous multimedia systems. Due to its wide range of application, such as human-computer interaction [21] and data-driven animation [19], automatic facial expression recognition has attracted significant attention in recent years. Recent facial expression recognition methods usually focus on extracting useful features and applying efficient classifiers such as neural-network-based methods [14], support vector machine (SVM) [13] and hidden Markov models (HMM) [27].

Although promising recognition results have been achieved, there still exist a common issue in previous facial expression recognition methods: the assumption that each facial image is associated with only one of the predefined affective labels tends to be an over-simplification. In real-world applications, a facial expression always contains blended emotions. For example, when one receives a letter from a friend whom he has not seen for a long time, he would be happy and surprised simultaneously. According to Plutchik's wheel of emotion theory [22], only a few emotions are basic emotions, and each facial expression usually expresses a mixture of basic emotions with different intensities. Therefore, to depict facial expressions more accurately, multi-label learning is utilized for facial expression recognition, and each picture is associated with multiple predefined emotions. For example, the GLMM (Group Lasso Regularized Maximum Margin) method was proposed to solve the facial expression recognition problem in the multi-label scenario [32]. However, there remain some cases that are not suitable to be solved by multi-label learning. Specially, in some cases, we need to know not only which emotions are associated with a facial expression but also the extent to which each emotion describes the expression. To solve such problems, label distribution learning (LDL) [6] is utilized to address facial expression recognition problems.

To the best of our knowledge, only one study [36] has been conducted on facial expression recognition by using LDL. Specifically, in this work, LDL was applied for facial

*Corresponding author: Zechao Li (zechao.li@njust.edu.cn). This work is jointly supported by National Natural Science Foundation of China (Grant No. 61773208), the Natural Science Foundation of Jiangsu Province (Grant No. BK20170809) and the China Postdoctoral Science Foundation (Grant No. 2018M632304).

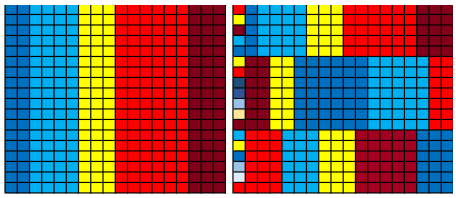


(a) (b) (c)

Figure 1: Illustration of non-global label correlations. *ANG* and *DIS* have a correlation in (a) and (b), but the correlation is not shared in (c).

expression recognition to improve the accuracy of facial expression recognition, and the label correlations are considered by seeking the Pearson's correlation coefficients [36]. Although their work attempted to exploit label correlations, it exploited label correlations in a global manne under the assumption that the correlations are shared by all instances. However, in real-world applications, label correlations are usually local, where a label correlation may be shared by only a subset of instances rather than all instances. For example, Fig. 1 gives three pictures from the *s-JAFFE* database with 6 basic emotions (happiness, sadness, surprise, anger, disgust and fear), and we consider the correlation between anger (*ANG*) and disgust (*DIS*). In Fig. 1a and Fig. 1b, *ANG* and *DIS* have similar description degrees in their respective images; thus, we consider that *ANG* and *DIS* have a correlation. However, in Fig. 1c, the description degree of *DIS* is significantly higher than *ANG*. Therefore, we deem that the correlation between *ANG* and *DIS* is not shared in Fig. 1c.

In this paper, we will solve the facial expression recognition problem by exploiting the emotion correlations at a local level, which has never been considered in previous LDL algorithms. Considering the complexity of emotion correlations, we adopt a low-rank structure to capture the local emotion correlations. Unlike previous works, we assume that the label space is the local low-rank structure shown in Fig. 2b rather than the global low-rank structure shown in Fig. 2a. As shown in Fig. 2b, it is not a low-rank structure at the global level but it can be divided into three blocks of low-rank structure. Based on this assumption, we propose an Emotion Distribution Learning method by exploiting Low-Rank label correlations Locally (EDL-LRL). Furthermore, we develop an alternating direction method of multipliers (ADMM) to optimize the objective function. Experiments on two widely used facial expression datasets show that our proposed method exhibits a promising performance when considering low-rank label correlations locally.



(a) global (b) local

Figure 2: A simple illustration of global and local label correlations. Each row denotes the label distribution of an instance, each column denotes an emotion and different colours denote different description degrees. The structure with same colour in each column constructs a low-rank structure that can capture the linear label correlation. (a) is the global low-rank structure and (b) is the local low-rank structure that can be divided into three blocks of low-rank structure.

The main contributions of this study can be summarized as follows: 1) different from the existing work that exploits the global label correlations, we consider the label correlations at a local leval; 2) unlike the existing work that calculate the pairwise label correlations explicitly, we employ a local low-rank structure to exploit the label correlations implicitly, which can capture the complex label correlations better. The remained of the paper is organized as follows. First, we briefly introduce facial expression recognition and label distribution learning. Second, we present the details of the proposed EDL-LRL algorithm. Finally, the experimental results are reported, followed by the conclusion.

2. Related Works

2.1. Facial Expression Recognition

Facial expressions are the facial changes in response to a person's internal emotional states, intentions, or social communications. Many studies have paid significant attention to facial expression recognition. Some approaches focused on feature extraction for the facial expression recognition problem, e.g., the shapes and locations of facial components are extracted to represent the face geometry [20]; action unit detection is presented by classifying features calculated from tracked fiducial facial points [26]. Moreover, other facial expression recognition research focused on applying different classifiers such as kNN [34], SVM [23] and Artificial Neural Networks(ANNs) [18]. Besides, emotions of facial expression were transmitted by some richer representations [2, 16] in recent years.

Although facial expression recognition methods have been designed from various perspectives, the goal of these works is to predict the most descriptive emotion from the

predefined affective labels. However, choosing only one emotion to represent the whole facial expression is inaccurate and insufficient because a facial expression usually contains a mixture of basic emotions with different intensities.

2.2. Label Distribution Learning

In recent years, learning with ambiguity has been a popular topic in machine learning area. There are three paradigms for solving label ambiguity at present, namely, single-label learning (SLL), multi-label learning (MLL) [25] and LDL. MLL has been successfully applied to facial expression recognition area [31]. Nevertheless, MLL cannot describe the extent of each label, in which it is unlikely that multiple affective labels have the same description degrees to the image. Thus, this paper describes a facial expression via an emotion distribution and employs LDL for prediction. LDL is a further extension of MLL, and LDL outputs a label distribution rather than a label set like MLL.

A number of algorithms, which can be divided into three groups, have been proposed for LDL. One group is based on the problem transformation (PT) strategy, which transforms an LDL problem into an SLL problem and changes the training examples to weighted single-label examples such as PT-SVM [9] and PT-Bayes [7]. The second group is based on the algorithm adaptation (AA). Certain algorithms, such as kNN and BP, are adapted to form the AA-kNN [8] and AA-BP [10], respectively. The final group consists of those based on the specialized algorithm (SA) that match the LDL problem directly such as SA-IIS [6] and SA-BFGS [6]. The related research has demonstrated that the third strategy is more effective than the other two strategies [6]. Therefore, this paper designs the emotion distribution learning algorithm based on the SA strategy as well.

To improve the performance of LDL, some algorithms attemp to exploit label correlations in different ways. In detail, the correlations were captured based on the Plutchik's wheel of emotions [35]; the label correlations were exploited by seeking the Pearson's correlation coefficients between two labels [36]; global label correlations were exploited for incomplete label distribution learning [28]; additional features were used to encode the influence of local sample correlations [33]; and a distance-mapping function was employed to encode the global label correlations [12]. However, these approaches exploited label correlations at a global level, and we explained that it is more reasonable to use the label correlations locally in the introduction.

3. Emotion Distribution Learning

3.1. Formalization

We will give a more formal definition of emotion distribution learning. Let $\mathcal{X} = R^q$ denote the q-dimensional image space of facial expressions, and let $\mathcal{Y} = \{y_1, y_2, \cdots, y_L\}$ denote the L predefined affective labels. Each label represents one of the basic emotions. Given a training set $S = \{(x_1, D_1), (x_2, D_2), \cdots, (x_n, D_n)\}$, where $D_i = \{d_i^1, d_i^2, \cdots, d_i^L\}$ is the emotion distribution with x_i, we assign a value d_i^j called the description degree to facial expression x_i for a particular emotion y_j, where $x_i \in \mathcal{X}$ and $y_j \in \mathcal{Y}$. Note that d_i^j is not the probability that y_j correctly labels x_i but rather is the proportion that y_j accounts for in a full description of x_i. All emotions with non-zero d_i^j-s are the correct emotions to describe the facial expression and satisfy $\sum_{j=1}^{L} d_i^j = 1$, which means that all emotions in the set can fully describe the facial expression. The goal of emotion distribution learning is to learn a mapping function $f : \mathcal{X} \to D$ that can predict the emotion distribution for unseen facial expression.

Suppose that $p(y|x;\theta)$ is the output model learnt from S, where θ is the parameter matrix. The goal of emotion distribution learning is to find an appropriate θ that can generate a distribution $p(y|x_i;\theta)$ similar to D_i given a facial expression x_i. Moreover, as for the form of $p(y|x_i;\theta)$, we assume it to be a maximum entropy model similar to previous work [6] as follows:

$$p(y_l|x_i;\theta) = \frac{1}{\mathcal{Z}_i} \exp(\sum_k \theta_{l,k} x_i^k), \tag{1}$$

where x_i^k is the k-th feature of x_i, $\theta_{l,k}$ is an element in θ, and $\mathcal{Z}_i = \sum_l \exp(\sum_k \theta_{l,k} x_i^k)$ is a normalization term used to satisfy the requirement that the sum of all emotion description degrees of an instance equals 1. In addition, we optimize θ by minimizing the following objective function, which incorporates global discrimination fitting and the influence of local label correlations:

$$\min_{\theta} V(\theta, S) + \lambda_1 \Omega(\theta, S) + \lambda_2 \Upsilon(\theta, S), \tag{2}$$

where V is the loss function defined on the training data, Ω is a regularizer to control the complexity of the output model, Υ is a regularizer to enforce the characteristic of local label correlations, and λ_1 and λ_2 are two parameters to balance the three terms.

With the previous discussion, the purpose of emotion distribution learning is to make the predicted distribution and the true distribution as similar as possible; therefore, we choose a loss function that can measure the similarity of two distributions. Various functions were analyzed to measure the similarity between two distributions such as the

Euclidean distance, Kullback-Leibler divergence and Jeffery divergence [4]. Here, for easy computation, we use the square of the Euclidean distance as the loss function defined by

$$D_J(Q_a||Q_b) = \sum_j (Q_a^j - Q_b^j)^2, \tag{3}$$

where Q_a^j and Q_b^j are the j-th element of the two distributions Q_a and Q_b, respectively. Specifically, in this paper, the expression for V based on the Euclidean distance is defined as follows:

$$V(\theta, S) = \frac{1}{2}\|D - \bar{D}\|_F^2, \tag{4}$$

where $\|\cdot\|_F^2$ denotes the Frobenius norm of a matrix, D and $\bar{D}$ denote the predicted distribution and the true distribution of the training set, respectively. For the second term of Eq. (2), we simply implement it as follows:

$$\Omega(\theta, S) = \|\theta\|_F^2. \tag{5}$$

The third term of Eq. (2) is employed to enforce the local low-rank structure of the predicted distribution, which implicitly exploits the label correlations locally. We assume that the training data can be divided into m clusters $\{G_1, G_2, \cdots, G_m\}$ and that each cluster is a low-rank structure. This partitioning can be implemented by clustering or some domain knowledge, such as gene pathways [24] and networks [5] in bioinformatics applications. For easy implementation, we use K-means as the clustering method. Notice that we cluster the training data in the label space rather than in the feature space because instances with similar label distributions usually share similar label correlations, and the cluster is more likely to be a low-rank structure. Unfortunately, the rank of a matrix is difficult to optimize; therefore, the trace norm $\|\cdot\|_{tr}$ is utilized in this paper as a convex approximation of the rank of a matrix. The trace norm $\|\cdot\|_{tr}$ is defined as the sum of singular values, i.e., $\|\cdot\|_{tr} = \sum_i \sigma_i(\cdot)$, where σ_i is the i-th singular value of the matrix. Thus, the final term of Eq. (2) based on local low-rank label correlations is derived as follows:

$$\Upsilon(\theta, S) = \sum_{i=1}^{m} \|D^{(i)}\|_{tr}, \tag{6}$$

where $D^{(i)}$ denotes the predicted distribution of the i-th cluster G_i. By substituting Eqs. (4), (5) and (6) into Eq. (2), the optimization problem is obtained as follows:

$$\min_{\theta} \frac{1}{2}\|D - \bar{D}\|_F^2 + \lambda_1\|\theta\|_F^2 + \lambda_2 \sum_{i=1}^{m} \|D^{(i)}\|_{tr}. \tag{7}$$

3.2. Optimizing using ADMM

ADMM (Alternating Direction Method of Multipliers) [3] is a simple but powerful algorithm that is well suited to solve Eq. (7). It takes the form of a decomposition-coordination procedure, in which the solutions to small local subproblems are coordinated to find a solution to a large global problem. For easy optimization in the following, we transform Eq. (7) into the form:

$$\begin{aligned} &\min_{\theta, Z} \frac{1}{2}\|D - \bar{D}\|_F^2 + \lambda_1\|\theta\|_F^2 + \lambda_2 \sum_{i=1}^{m} \|Z^{(i)}\|_{tr} \\ &s.t. \quad D^{(i)} - Z^{(i)} = 0. \end{aligned} \tag{8}$$

The augmented Lagrange function of Eq. (8) is given by

$$\begin{aligned} &\min_{\theta, Z, \Lambda} \frac{1}{2}\|D - \bar{D}\|_F^2 + \lambda_1\|\theta\|_F^2 + \lambda_2 \sum_{i=1}^{m} \|Z^{(i)}\|_{tr} \\ &+ \sum_{i=1}^{m} < \Lambda^{(i)}, D^{(i)} - Z^{(i)} > + \sum_{i=1}^{m} \frac{\rho^{(i)}}{2}\|D^{(i)} - Z^{(i)}\|_F^2, \end{aligned} \tag{9}$$

where Λ is a list of Lagrange multipliers, consisting of $\{\Lambda^{(1)}, \Lambda^{(2)}, \cdots, \Lambda^{(m)}\}$; ρ is a list of positive numbers, called the penalty parameters, consisting of $\{\rho^{(1)}, \rho^{(2)}, \cdots, \rho^{(m)}\}$; Z consists of $\{Z^{(1)}, Z^{(2)}, \cdots, Z^{(m)}\}$; and $< \cdot, \cdot >$ is the Frobenius dot-product, i.e., for two matrices $X, Y \in R^{m \times n}$, $< X, Y >= tr(X^T Y) = \sum_{i=1}^{m}\sum_{j=1}^{n} X_{ij} Y_{ij}$. Now, the above optimization problem can be solved by alternating minimization, i.e., update each variable (θ, Z and Λ) with the others fixed in iteration t:

$$\begin{aligned} \theta^{t+1} = &\arg\min_{\theta} \frac{1}{2}\|D - \bar{D}\|_F^2 + \lambda_1\|\theta\|_F^2 \\ &+ \sum_{i=1}^{m} < \Lambda^{(i)t}, D^{(i)} - Z^{(i)t} > \\ &+ \sum_{i=1}^{m} \frac{\rho^{(i)}}{2}\|D^{(i)} - Z^{(i)t}\|_F^2, \end{aligned} \tag{10}$$

$$\begin{aligned} Z^{t+1} = &\arg\min_{Z} \lambda_2 \sum_{i=1}^{m} \|Z^{(i)}\|_{tr} \\ &+ \sum_{i=1}^{m} < \Lambda^{(i)t}, D^{(i)t+1} - Z^{(i)} > \\ &+ \sum_{i=1}^{m} \frac{\rho^{(i)}}{2}\|D^{(i)t+1} - Z^{(i)}\|_F^2, \end{aligned} \tag{11}$$

$$\Lambda^{(i)t+1} = \Lambda^{(i)t} + \rho^{(i)}(D^{(i)t+1} - Z^{(i)t+1}). \tag{12}$$

Eq. (10) can be effectively solved by the limited-memory quasi-Newton method (L-BFGS) [30]. The basic idea is to avoid explicit calculation of the inverse Hessian matrix used in the Newton method. In addition, L-BFGS approximates the inverse Hessian matrix with an iteratively updated matrix instead of storing the full matrix. Let Eq. (10)

be $T(\theta)$, we follow the idea of an effective quasi-Newton method BFGS. Consider the second order Taylor series of $T'(\theta) = -T(\theta)$ at the current estimate of the parameter vector $\theta^{(l)}$:

$$T'(\theta^{(l+1)}) \approx T'(\theta^{(l)}) + \nabla T'(\theta^{(l+1)})^T \Delta + \frac{1}{2}\Delta^T H(\theta^{(l)})\Delta, \tag{13}$$

where $\Delta = \theta^{(l+1)} - \theta^{(l)}$ is the update step, $\nabla T(\theta^{(l)})$ and $H(\theta^{(l)})$ are the gradient and Hessian matrix of $T'(\theta^{(l+1)})$ at $\theta^{(l)}$, respectively. The minimizer of Eq. (13) is

$$\Delta^{(l)} = -H^{-1}(\theta^{(l)})\nabla T'(\theta^{(l)}). \tag{14}$$

The line search Newton method uses $\Delta^{(l)}$ as the search direction $p^{(l)} = \Delta^{(l)}$ and updates model parameters by

$$\theta^{(l+1)} = \theta^{(l)} + \alpha^{(l)} p^{(l)}, \tag{15}$$

where the step length $\alpha^{(l)}$ is obtained from a line search procedure to satisfy the strong Wolfe conditions [17]:

$$T'(\theta^{(l)} + \alpha^{(l)} p^{(l)}) \leqslant T'(\theta^{(l)}) + c_1 \alpha^{(l)} \nabla T'(\theta^{(l)})^T p^{(l)}, \tag{16}$$

$$|\nabla T'(\theta^{(l)} + \alpha^{(l)} p^{(l)})| \leqslant c_2 |\nabla T'(\theta^{(l)})^T p^{(l)}|, \tag{17}$$

where $0 < c_1 < c_2 < 1$. The idea of L-BFGS is to avoid explicit calculation of $H^{-1}(\theta^{(l)})$ by approximating it with an iteratively updated matrix B, i.e.

$$\begin{aligned} B^{(l+1)} = &(I - \rho^{(l)} s^{(l)} (u^{(l)})^T) B^{(l)} (I - \rho^{(l)} u^{(l)} (s^{(l)})^T) \\ &+ \rho^{(l)} s^{(l)} (s^{(l)})^T, \end{aligned} \tag{18}$$

where $s^{(l)} = \theta^{(l+1)} - \theta^{(l)}$, $u^{(l)} = \nabla T'(\theta^{(l+1)}) - \nabla T'(\theta^{(l)})$ and $\rho^{(l)} = \frac{1}{s^{(l)} u^{(l)}}$.

As for the optimization of Eq. (10), the computation of L-BFGS is mainly related to the first-order gradient, which can be obtained by

$$\begin{aligned} \nabla \theta_{l,k} = &\sum_{i=1}^{n} (p(y_l|x_i;\theta) - d_i^l) p'(y_l|x_i;\theta) + 2\lambda_1 \theta_{l,k} \\ &+ \sum_{i=1}^{m} \sum_{x_j \in G_i} \Lambda_{j,l}^{(i)} p'(y_l|x_j;\theta) \\ &+ \sum_{i=1}^{m} \sum_{x_j \in G_i} \rho^{(i)} (p(y_l|x_j;\theta) - Z_j^{(i)l}) p'(y_l|x_j;\theta), \end{aligned} \tag{19}$$

where $\Lambda_{j,l}^{(i)}$ is an element of $\Lambda^{(i)}$, $Z_j^{(i)l}$ is an element of $Z^{(i)}$ and $p'(y_l|x_i;\theta) = x_i^k (p(y_l|x_i;\theta) - p^2(y_l|x_i;\theta))$.

To solve Eq. (11), it can be decomposed into m optimization problems, where the i-th problem is:

$$\begin{aligned} \min_{Z^{(i)}} &\lambda_2 \|Z^{(i)}\|_{tr} + < \Lambda^{(i)t}, D^{(i)t+1} - Z^{(i)} > \\ &+ \frac{\rho^{(i)}}{2} \|D^{(i)t+1} - Z^{(i)}\|_F^2. \end{aligned} \tag{20}$$

Algorithm 1: The EDL-LRL algorithm

Input: training set $S = \{X, D\}$, parameters λ_1, λ_2 and m.
Output: the label distribution D_t.

```
cluster training set S with K-means;
initialize Λ, Z, ρ and θ;
t = 1;
repeat
    solve θ^{t+1} by Eq. (10);
    solve Z^{t+1} by Eq. (11);
    update Λ^{t+1} by Eq. (12);
    t = t + 1;
until stopping criterion is satisfied;
return the label distribution D_t according to Eq. (1).
```

Then, Eq. (20) can be further rewritten as follows:

$$\min_{Z^{(i)}} \frac{\lambda_2}{\rho^{(i)}} \|Z^{(i)}\|_{tr} + \frac{1}{2} \|Z^{(i)} - (D^{(i)} + \frac{\Lambda^{(i)}}{\rho^i})\|_F^2, \tag{21}$$

which has closed-form solutions. Eq. (21) can be solved by the following Lemma 1:

Lemma 1 *For matrix $Y \in R^{n \times d}$ and $\mu > 0$, the problem as follows has the only one analysis solution,*

$$\arg \min_{M \in R^{n \times d}} \mu \|M\|_{tr} + \frac{1}{2} \|M - Y\|_F^2.$$

This solution can be described by singular value thresholding operator,

$$SVT_\mu(Y) = U diag[(\sigma - \mu)_+] V^T$$

$$(\sigma - \mu)_+ = \begin{cases} \sigma - \mu & \sigma > \mu \\ 0 & otherwise, \end{cases}$$

$U \in R^{n \times r}$, $V \in R^{d \times r}$ and $\sigma = \{\sigma_1, \sigma_2, \cdots, \sigma_r\} \in R^{r \times 1}$ can be achieved by singular decomposition of matrix Y, $Y = U \Sigma V^T$ and $\Sigma = diag(\sigma)$.

The overall procedure of our proposed algorithm is presented in Algorithm 1. Moreover, the ADMM method in our algorithm will converge at $O(1/T)$ rate to the optimum solution according to [11], where T is the number of iteration steps. Although it can be slower to converge to high accuracy than other optimization methods, a modest accuracy is sufficient to attain satisfactory performance [3].

4. Experiments

4.1. Datasets

Various datasets are widely used in the facial expression recognition area. However, most of them are only suitable

Dateset	Examples	Features	Lables
s-JAFFE	213	243	6
SBU_3DFE	2500	243	6

Table 1: The characteristics of two datasets.

for single-emotion or multi-emotion problems rather than the emotion distribution problem. While our proposed approach prefers to the distribution datasets with annotations of different voters, the majority voting scheme is widely adopted as the ground truth in this area. Unfortunately, there are few facial expression datasets provide the detailed votes from all the workers. Therefore, to evaluate the effectiveness of our proposed algorithm, we performed extensive experiments on two facial expression datasets, *s-JAFFE* and *SBU_3DFE*, which are extended from JAFFE [15] and BU_3DFE [29], respectively. The characteristics of the two datasets are summarized in Table 1.

The *s-JAFFE* dataset contains 213 grayscale images of 10 Japanese female models. Each image is scored by 60 people on the 6 basic emotions (i.e., happiness, sadness, surprise, fear, anger and disgust) with a five-level scale (1 represents the lowest emotion intensity, while 5 represents the highest emotion intensity). The average score (after normalization) of each emotion is used to represent the emotion distribution. The second dataset, named *SBU_3DFE*, contains 2500 images, and the emotion distribution of each image is obtained by the same method as *s-JAFFE*. Besides, a 243-dimensional feature vector is extracted from each image in *s-JAFFE* and *SBU_3DFE* by Local Binary Patterns (LBP) method [1].

4.2. Evaluation Measures

Different from [36], a different set of measures are used in our paper, because our used measures are more representative that are validated in [6]. In detail, six measures, including distance-based measures and similarity-based measures [6], are chosen as the evaluation measures for the LDL algorithms in this paper. The names and formulas are presented in Table 2, where $D = \{d_1, d_2, \cdots, d_L\}$ denote the predicted label distribution and $\bar{D} = \{\bar{d}_1, \bar{d}_2, \cdots, \bar{d}_L\}$ denote the real label distribution. For the first four distance measures, "↓" indicates "the smaller, the better", and "↑" indicates "the larger, the better" for the last two similarity measures.

4.3. Experimental Setting

To verify the performance of the proposed EDL-LRL method, we take PT-SVM [9], PT-Bayes [7], AA-kNN [8], AA-BP [10], SA-IIS [6], SA-BFGS [6], EDL [36], LDL-SCL [33] and LDLLC [12] in our comparison. The parameter settings of those algorithms are as follows. PT-SVM is implemented as the "C-SVC" type in LIBSVM using the RBF kernel with the parameters $C = 1.0$ and $Gamma = 0.01$. For PT-Bayes, maximum likelihood estimation is employed to estimate the Gaussian class-conditional probability density functions. The number of neighbors k in AA-kNN is set to 5 and the number of hidden-layer neurons for AA-BP is set to 60. The parameters in SA-BFGS are set to: $c_1 = 10^{-4}$ and $c_2 = 0.9$. The parameters η, ε, ξ_1 and ξ_2 in EDL are set as 5, 0.25, 0.0001, 0.001, respectively. For LDL-SCL, λ_1, λ_2 and λ_3 are set to 0.001. For LDLLC, the parameters are set to: $\lambda_1 = 0.1$ and $\lambda_2 = 0.01$. In addition, for EDL-LRL, the regularization parameters λ_1 and λ_2 are set as 10^{-3} and 10^{-2}, respectively. The number of clusters obtained by K-means is set to 5, i.e., $m = 5$, and we will investigate the influence of m in the following.

	Name	Formula
Distance	Chebyshev↓	$Dis_1(\bar{D}, D) = \max_j \lvert\bar{d}_j - d_j\rvert$
	Clark↓	$Dis_2(\bar{D}, D) = \sqrt{\sum_{j=1}^{L} \frac{(\bar{d}_j - d_j)^2}{(\bar{d}_j + d_j)^2}}$
	Canberra ↓	$Dis_3(\bar{D}, D) = \sum_{j=1}^{L} \frac{\lvert\bar{d}_j - d_j\rvert}{\bar{d}_j + d_j}$
	Kullback-Leibler(K-L)↓	$Dis_4(\bar{D}, D) = \sum_{j=1}^{L} \bar{d}_j \ln \frac{\bar{d}_j}{d_j}$
Similarity	Cosine↑	$Sim_1(\bar{D}, D) = \frac{\sum_{j=1}^{L} \bar{d}_j d_j}{\sqrt{\sum_{j=1}^{L} \bar{d}_j^2}\sqrt{\sum_{j=1}^{L} d_j^2}}$
	Intersection↑	$Sim_2(\bar{D}, D) = \sum_{j=1}^{L} \min(\bar{d}_j, d_j)$

Table 2: Evaluation measures for LDL algorithms.

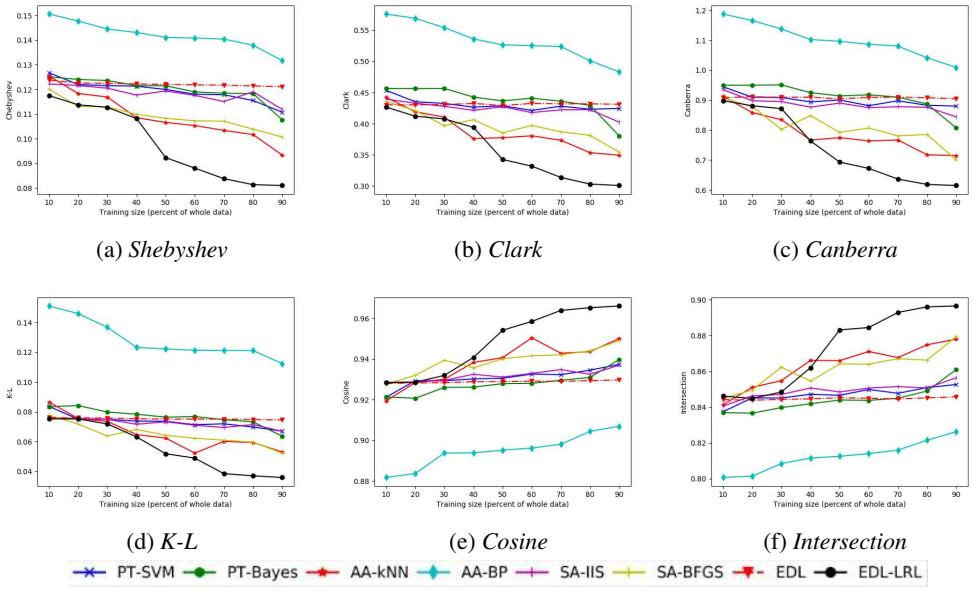
4.4. Results and Discussion

For each dataset, the five-fold cross validation is employed in this paper. In detail, the instances in each dataset are randomly divided into 5 parts, one part for testing and the remainder for training. Note that the EDL results compared in our experiments are different with those reported in [36], because they used 90% instances as the training set whereas we use 80% instances, since a small test set is difficult to reflect the difference between different algorithms (For *s-JAFFE*, 10% instances has only 21 instances). We apply each method 10 times on each dataset, and the experimental results are presented in the form of "mean± std". The experimental results are reported in Table 3. The best performance on each measure is marked in bold, and the two-tailed t-test with 5% significance level is performed to see whether the differences between our method (EDL-LRL) and the other methods are statistically significant. The results of the t-test are presented immediately after the performance of each method, where • (○) indicates significance difference.

As illustrated in Table 3, our proposed EDL-LRL method outperforms all other methods (PT-SVM [9], PT-Bayes [7], AA-kNN [8], AA-BP [10], SA-IIS [6], SA-

data	algorithm	Chebyshev↓	Clark↓	Canberra↓	K-L↓	Cosine↑	Intersection↑
s-JAFFE	EDL-LRL	**0.0806±0.006**	**0.3008±0.016**	**0.6134±0.034**	**0.0361±0.004**	**0.9660±0.004**	**0.8970±0.006**
	PT-SVM	0.1238±0.027•	0.4353±0.045•	0.9039±0.099•	0.0745±0.024•	0.9290±0.023•	0.8453±0.021•
	PT-Bayes	0.1204±0.030•	0.4287±0.053•	0.8972±0.138•	0.0764±0.031•	0.9287±0.027•	0.8470±0.029•
	AA-kNN	0.1009±0.015•	0.3562±0.036•	0.7283±0.102•	0.0527±0.018•	0.9484±0.020•	0.8739±0.018•
	AA-BP	0.1447±0.015•	0.5330±0.062•	1.0995±0.109•	0.1180±0.027•	0.8959±0.021•	0.8132±0.017•
	SA-IIS	0.1202±0.048•	0.4651±0.050•	0.9349±0.125•	0.0775±0.036•	0.9286±0.036•	0.8442±0.031•
	SA-BFGS	0.1007±0.029•	0.3847±0.094•	0.7825±0.195•	0.0568±0.024•	0.9452±0.025•	0.8673±0.032•
	EDL	0.1211±0.008•	0.4311±0.022•	0.9050±0.054•	0.0745±0.008•	0.9297±0.007•	0.8458±0.010•
	LDL-SCL	0.0890±0.007•	0.3304±0.021•	0.6808±0.047•	0.0443±0.006•	0.9583±0.006•	0.8851±0.009•
	LDLLC	0.1194±0.010•	0.4207±0.016•	0.8775±0.043•	0.0713±0.008•	0.9324±0.008•	0.8503±0.009•
SBU_3DFE	EDL-LRL	**0.0951±0.002**	**0.3556±0.006**	**0.7463±0.013**	0.0694±0.002	**0.9626±0.002**	**0.8686±0.002**
	PT-SVM	0.1439±0.006•	0.4305±0.012•	0.9321±0.027•	0.0926±0.008•	0.9113±0.007•	0.8324±0.005•
	PT-Bayes	0.1451±0.005•	0.4292±0.013•	0.9413±0.031•	0.0904±0.005•	0.9126±0.004•	0.8314±0.005•
	AA-kNN	0.1300±0.004•	0.4105±0.005•	0.8532±0.015	0.0845±0.007•	0.9176±0.005•	0.8453±0.003•
	AA-BP	0.1475±0.004•	0.4925±0.031•	1.0345±0.063•	0.1205±0.042•	0.8952±0.011•	0.8158±0.008•
	SA-IIS	0.1405±0.005•	0.4270±0.016•	0.9241±0.038•	0.0852±0.007•	0.9166±0.005•	0.8341±0.006•
	SA-BFGS	0.1291±0.009•	0.3984±0.016•	0.8596±0.046•	0.0758±0.008•	0.9255±0.007•	0.8454±0.008•
	EDL	0.1377±0.002•	0.4099±0.003•	0.8970±0.007•	0.0844±0.001•	0.9185±0.001•	0.8397±0.001•
	LDL-SCL	0.1106±0.002•	0.3749±0.003•	0.7517±0.007	**0.0574±0.001**○	0.9435±0.001•	0.8656±0.001•
	LDLLC	0.1356±0.003•	0.4328±0.007•	0.9290±0.015•	0.0857±0.003•	0.9165±0.002•	0.8330±0.003•

Table 3: Comparison results (mean±std.) of LDL methods on real-world datasets. The best performance on each measure is marked in bold. • (○) indicates that EDL-LRL is significantly better (worse) than the corresponding method on the criterion based on two-tailed t-test with 5% significance level. ↑ (↓) indicates the larger (smaller), the better.



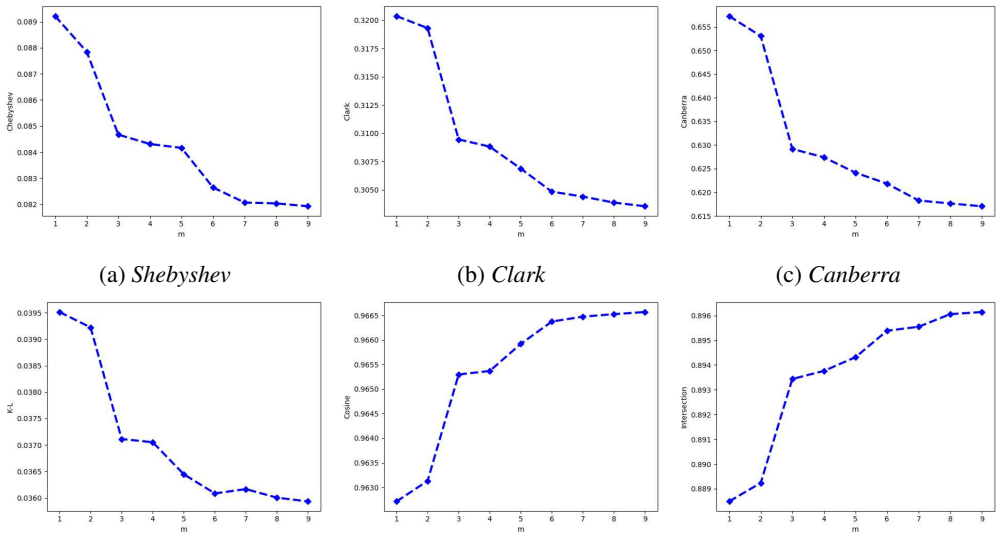
(a) *Shebyshev* (b) *Clark* (c) *Canberra*

(d) *K-L* (e) *Cosine* (f) *Intersection*

Figure 3: Comparison of eight methods under varying the training data sizes on *s-JAFFE*.

BFGS [6], EDL [36], LDL-SCL [33] and LDLLC [12]) on all criteria except for *K-L*, and EDL-LRL has the top 2 performances on *K-L*. Besides, the specialized LDL algorithms generally perform better than those algorithms obtained from PT and AA in most cases. The reason is that the specialized LDL algorithms are designed to directly minimize the similarity between the predicted label distribution and the true label distribution. Furthermore, it is worth mentioning that the EDL-LRL method is superior to the EDL and LDLLC methods, which exploit label correlations in a global manner. This indicates that exploiting label correlations locally is more reasonable.

In addition, to demonstrate the robustness of our proposed method, we studied the performance of emotion distribution prediction under varying training data sizes. In the experiment, $10\% - 90\%$ of the data are used as the training set. A desired number of instances are sampled randomly ten times, and the resulting average *Chebyshev*, *Clark*, *Can-*



(a) *Shebyshev* (b) *Clark* (c) *Canberra*

(d) *K-L* (e) *Cosine* (f) *Intersection*

Figure 4: Influence of m with 6 measures on dataset *s-JAFFE*.

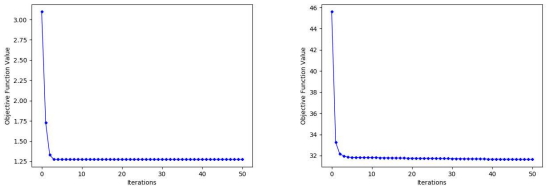
berra, *K-L*, *Cosine* and *Intersection* are recorded. We report the experimental results on the *s-JAFFE* dataset with 6 evaluation measures, which are shown in Fig. 3. For the simplicity of illustration and the poor performances of PT-SVM and AA-BP, the results of these algorithms are not reported in Fig. 3. It can be observed that the performance of our EDL-LRL method improves as the training data size increases. Furthermore, EDL-LRL achieves the best performance when the training data size is greater than 40%, and it is in the top 3 performances when the training data size is less than 40% because the local label correlations cannot be effectively exploited when the training set is insufficient.

4.5. Influence of the Number of Clusters

To investigate the influence of the number of clusters m, we run EDL-LRL with m varying from 1 to 9, and five-fold cross validation is employed in each experiment. Besides, we only show the results with six measures on the *s-JAFFE* dataset because the results have the similar trend on the *SBU_3DFE*. As seen in Fig. 4, the performance improves as m increases and tends to be stable after m becomes sufficiently large.

4.6. Convergence

An ADMM based optimization method is used to solve the objective function of our algorithm. To investigate the convergence of the ADMM method to solve the EDL-LRL model, we plot the value of the objective function (i.e., Eq. (8)) on the two datasets in Fig. 5. As can be observed, the objective function value decreases with respect to the number of iterations, and the value approaches a fixed value after a few iterations.



(a) *s-JAFFE* (b) *SBU_3DFE*

Figure 5: Convergence of EDL-LRL on *s-JAFFE* and *SBU_3DFE*.

5. Conclusion

To depict facial expressions more accurately, this paper introduces a challenging learning scenario wherein facial expression recognition is modeled as an emotion distribution learning problem and proposes the EDL-LRL algorithm. Moreover, we exploit the label correlations at a local level since different facial expressions may share different label correlations in real-world applications, and a local low-rank assumption is employed to capture the local label correlations. A series of experiments demonstrate that EDL-LRL is superior to some state-of-the-art label distribution learning methods.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition.

IEEE Trans. Pattern Anal. Mach. Intell., 28(12):2037–2041, 2006.

[2] Carlos F. Benitezquiroz, Ramprakash Srinivasan, and Aleix M Martinez. Facial color is an efficient mechanism to visually transmit emotion. *Proceedings of the National Academy of Sciences of the United States of America*, 115(14):3581–3586, 2018.

[3] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations & Trends in Machine Learning*, 3(1):1–122, 2011.

[4] Sung Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models & Methods in Applied Sciences*, 1(4):300–307, 2007.

[5] Hanyu Chuang, Eunjung Lee, Yutsueng Liu, Doheon Lee, and Trey Ideker. Network-based classification of breast cancer metastasis. *Molecular Systems Biology*, 3(1):140–149, 2007.

[6] Xin Geng. Label distribution learning. *IEEE Transactions on Knowledge & Data Engineering*, 28(7):1734–1748, 2016.

[7] Xin Geng and Rongzi Ji. Label distribution learning. In *IEEE International Conference on Data Mining Workshops*, pages 377–383, 2013.

[8] Xin Geng, Kate Smith-Miles, and Zhihua Zhou. Facial age estimation by learning from label distributions. In *AAAI Conference on Artificial Intelligence*, pages 451–456, 2010.

[9] Xin Geng, Qin Wang, and Yu Xia. Facial age estimation by adaptive label distribution learning. In *IEEE International Conference on Pattern Recognition*, pages 4465–4470, 2014.

[10] Xin Geng, Chao Yin, and Zhihua Zhou. Facial age estimation by learning from label distributions. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 35(10):2401–2412, 2013.

[11] Bingsheng He and Xiaoming Yuan. On the $O(1/n)$ convergence rate of the douglas-rachford alternating direction method. *SIAM Journal on Numerical Analysis*, 50(2):700–709, 2012.

[12] Xiuyi Jia, Weiwei Li, Junyu Liu, and Yu Zhang. Label distribution learning by exploiting label correlations. In *AAAI Conference on Artificial Intelligence*, pages 3310–3317, 2018.

[13] Irene Kotsia and Ioannis Pitas. Facial expression recognition in image sequences using geometric deformation features and support vector machines. *IEEE Transactions on Image Processing*, 16(1):172–187, 2006.

[14] Ping Liu, Shizhong Han, Zibo Meng, and Yan Tong. Facial expression recognition via a boosted deep belief network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1805–1812, 2014.

[15] M Lyons, S Akamatsu, M Kamachi, and J Gyoba. Coding facial expressions with gabor wavelets. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 200–205, 2002.

[16] Aleix M Martinez. Visual perception of facial expressions of emotion. *Current Opinion in Psychology*, 17:27–33, 2017.

[17] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Science & Business Media, 2006.

[18] Ebenezer Owusu, Yongzhao Zhan, and Qirong Mao. A neural-adaboost based facial expression recognition system. *Expert Systems with Applications*, 41(7):3383–3390, 2014.

[19] Maja Pantic and Leon J M Rothkrantz. Automatic analysis of facial expressions: the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1424–1445, 2000.

[20] M. Pantic and L. J. M Rothkrantz. Expert system for automatic analysis of facial expressions. *Image & Vision Computing*, 18(11):881–905, 2000.

[21] Maja Pantic and Leon J M Rothkrantz. Toward an affect-sensitive multimodal human-computer interaction. *Proceedings of the IEEE*, 91(9):1370–1390, 2003.

[22] Robert Plutchik. *Chapter 1 - A General Psychoevolutionary Theory of Emotion*. Elsevier Inc., 1980.

[23] Caifeng Song, Weifeng Liu, and Yanjiang Wang. Facial expression recognition based on hessian regularized support vector machine. In *International Conference on Internet Multimedia Computing and Service*, pages 264–267, 2013.

[24] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda G Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, 2005.

[25] Grigorios Tsoumakas, Ioannis Katakis, and David Taniar. Multi-label classification: An overview. *International Journal of Data Warehousing & Mining*, 3(3):1–13, 2007.

[26] M. Valstar, I. Patras, and M. Pantic. Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, pages 76–84, 2005.

[27] Te Hsun Wang and Jenn Jier James Lien. Facial expression recognition system based on rigid and non-rigid motion separation and 3D pose estimation. *Pattern Recognition*, 42(5):962–977, 2009.

[28] Miao Xu and Zhihua Zhou. Incomplete label distribution learning. In *International Joint Conference on Artificial Intelligence*, pages 3175–3181, 2017.

[29] Lijun Yin, Xiaozhou Wei, Yi Sun, Jun Wang, and Matthew J Rosato. A 3D facial expression database for facial behavior research. In *International Conference on Automatic Face and Gesture Recognition*, pages 211–216, 2006.

[30] Yaxiang Yuan. A modified BFGS algorithm for unconstrained optimization. *IMA Journal of Numerical Analysis*, 11(3):325–332, 1991.

[31] Kaili Zhao, Wensheng Chu, Fernando De la Torre, Jeffrey F. Cohn, and Honggang Zhang. Joint patch and multi-label learning for facial action unit and holistic expression recognition. *IEEE Transactions on Image Processing*, 25(8):3931–3946, 2016.

[32] Kaili Zhao, Honggang Zhang, Mingzhi Dong, Jun Guo, Yonggang Qi, and Yizhe Song. A multi-label classification approach for facial expression recognition. In *Visual Communications and Image Processing*, pages 1–6, 2014.

[33] Xiang Zheng, Xiuyi Jia, and Weiwei Li. Label distribution learning by exploiting sample correlations locally. In *AAAI Conference on Artificial Intelligence*, pages 4556–4563, 2018.

[34] Ruicong Zhi and Qiuqi Ruan. Facial expression recognition based on two-dimensional discriminant locality preserving projections. *Neurocomputing*, 71(79):1730–1734, 2008.

[35] Deyu Zhou, Xuan Zhang, Yin Zhou, Quan Zhao, and Xin Geng. Emotion distribution learning from texts. In *Conference on Empirical Methods in Natural Language Processing*, pages 638–647, 2016.

[36] Ying Zhou, Hui Xue, and Xin Geng. Emotion distribution recognition from facial expressions. In *ACM International Conference on Multimedia*, pages 1247–1250, 2015.