Pattern Recognition 47 (2014) 1859–1868

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Spontaneous facial expression recognition: A robust metric learning approach

Shaohua Wan *, J.K. Aggarwal [1]

Computer Vision Research Center, The University of Texas at Austin, Austin, TX 78712-1084, US

ARTICLE INFO

Article history:
Received 12 July 2013
Received in revised form
9 October 2013
Accepted 23 November 2013
Available online 3 December 2013

Keywords:
Spontaneous facial expression recognition
Metric learning
Online learning
Robust learning

ABSTRACT

Spontaneous facial expression recognition is significantly more challenging than recognizing posed ones. We focus on two issues that are still under-addressed in this area. First, due to the inherent subtlety, the geometric and appearance features of spontaneous expressions tend to overlap with each other, making it hard for classifiers to find effective separation boundaries. Second, the training set usually contains dubious class labels which can hurt the recognition performance if no countermeasure is taken. In this paper, we propose a spontaneous expression recognition method based on robust metric learning with the aim of alleviating these two problems. In particular, to increase the discrimination of different facial expressions, we learn a new metric space in which spatially close data points have a higher probability of being in the same class. In addition, instead of using the noisy labels directly for metric learning, we define sensitivity and specificity to characterize the annotation reliability of each annotator. Then the distance metric and annotators' reliability is jointly estimated by maximizing the likelihood of the observed class labels. With the introduction of latent variables representing the true class labels, the distance metric and annotators' reliability can be iteratively solved under the Expectation Maximization framework. Comparative experiments show that our method achieves better recognition accuracy on spontaneous expression recognition, and the learned metric can be reliably transferred to recognize posed expressions.

1. Introduction

Human emotion recognition has long been an actively researched topic in Human Computer Interaction (HCI). Unlike other types of non-verbal communication, the human face is expressive and closely tied to an emotional state. The ability to interpret facial gestures is a key to a wide range of HCI applications. Researchers have achieved tremendous success in recognizing prototypical and posed facial expressions that are collected under tightly controlled conditions [1–3]. Since the most useful current and future face related applications lie in a more natural context, it is our goal in this paper to develop a system that can operate on spontaneous expressions characterizing the natural interaction between humans and computers.

Quite a few studies have been done on spontaneous facial expression recognition [4–6], but with only limited progress. There are several factors affecting the recognition accuracy of spontaneous expressions, including facial feature representation, classifier design, useful contextual cues, etc. This paper focuses on two issues that are still under-addressed in this field.

First of all, spontaneous facial expressions tend to have overlapping geometric and appearance features, making it difficult to find effective classification boundaries [6]. The second issue, most often ignored, has to do with noisy labeling. Traditional supervised classification methods assume perfect data labels. However, in the case of spontaneous facial expression recognition, which involves only slight facial muscle actions, the class labels can be erroneously assigned due to the subjectivity or varied expertise of the annotators [7]. Classifiers trained on such data inevitably have their performance negatively affected.

In this paper, we present an automatic recognition system for spontaneous facial expressions. In particular, we make the following contributions.

First, we formulate spontaneous facial expression recognition as a maximum likelihood based metric learning problem. Under the learned distance metric, spatially close (distant) data points have a higher probability of being in the same class, thus facilitating the kNN based classification.

Second, we address the problem of noisy labeling via multi-annotation and reliability estimation. In particular, to increase robustness to noisy labels, for each data point, multiple labels from different annotators are collected. The sensitivity and specificity

* Corresponding author. Permanent address: 3541 N Hills. Dr, Austin, TX 78731, US. Tel.: +1 512 363 9242; fax: +1 512 471 5532.
E-mail addresses: shaohuawan@gmail.com, shaohuawan@utexas.edu (S. Wan), aggarwaljk@mail.utexas.edu (J.K. Aggarwal).
[1] Tel.: +1 512 471 1369.

0031-3203/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.11.025

of each annotator, which indicates the annotation reliability, and the distance metric is jointly estimated under the Expectation Maximization (EM) framework via an efficient online learning algorithm.

Third, we extensively compare our method with other methods. Experiments show that our method not only performs significantly better in recognizing spontaneous expressions, but also generalizes well to posed expressions.

The rest of this paper is structured as follows. In Section 2, a brief review of related work is given. In Section 3, the problem setting is described. Section 4 describes the feature representation of a facial expression. We formulate the problem of Robust Metric Learning based expression recognition and give an efficient solution in Section 5. Experimental results are given in Section 6.

2. Related work

Facial expression recognition methods are usually concerned with 7 basic expressions (including neutral) as defined in [8], and may be broadly classified as static or dynamic. Static approaches classify an expression in a single static image without considering the contextual information implied by adjacent images of a sequence. Representative methods are Naive Bayesian [9], SVM [1], Adaboost [10], etc. In contrast, a dynamic approach, e.g. HHM [11], CRF [12], exploits the dependency between adjacent images of a sequence to boost the recognition accuracy. Most of these studies have focused on posed expression recognition. For a comprehensive survey, we refer readers to [13].

Recently, there has been a shift of research interest from posed to spontaneous expressions. Valstar et al. [4] show that the temporal modeling of brow actions is important for recognizing spontaneous expressions. In Cohn and Schmidt's work [5], the amplitude and timing of facial movement are shown to be important in recognizing spontaneous smiles. Park et al. [14] magnify subtle expressions using face-region-specific weight factors to obtain new feature representation. Note that these methods only deal with spontaneous expressions on a small scale (a limited number of examples and expression classes), and none of them explicitly model the inter-class overlapping and noisy labeling. Ref. [6] is similar to this work by employing a learned metric for nearest-neighbor expression retrieval, but it fails to accommodate noisy labels in the training data. For a comprehensive review of spontaneous facial expression recognition, we refer readers to [15].

The facial features used in most previous work are either geometric features such as the shape of the facial components (eyes, mouth, etc.) [12,16,17] or appearance features representing facial texture (wrinkles, bulges, furrows, etc.) [18–20]. As suggested in several studies, e.g. [21], combining both geometric and appearance features is a better choice in representing facial expressions.

However, due to the subtlety of spontaneous expressions, the geometric and appearance features of different classes tend to overlap with each other [6]. Our work is based on the idea that spontaneous expressions can be better recognized in a learned feature space where spatially close (distant) data points have a higher probability of being in the same (different) class.

In particular, let the distance metric of the learned feature space be denoted by M. The distance between a pair of expressions (x^1, x^2), which is parameterized by M, can be written as

$$d_M(x^1, x^2) = (x^1 - x^2)^T M (x^1 - x^2) \quad (1)$$

From the perspective of metric learning, M should be such that expressions within the same class are closer to each other; otherwise they are separated by a large margin. Various methods have been proposed to learn the metric, with different objective functions designed for specific tasks (clustering [22], classification [23–25]).

Compared to the previous metric learning methods, ours has a direct probabilistic interpretation that the likelihood of two expressions being in the same class is modeled as a sigmoid function of their distance in the learned feature space. The resulting optimization is solved as a maximum likelihood estimation problem under the Expectation Maximization framework. Guillaumin et al. [26] use a similar maximum likelihood metric learning formulation for face recognition. However, they do not enforce the metric to be positive semidefinite. Hence, the learned feature space may be such that the distance between data points does not satisfy symmetry and triangle inequality.

As mentioned in previous section, the labels of the spontaneous facial expressions used for training can be quite noisy. There have been several notable works addressing the issue of learning from samples with noisy labeling. Opper et al. [27] propose to weaken the confidence of the observed class labels via a scaling factor. This method is easy to implement but has only limited accuracy without considering the underlying label flipping process. Xue et al. [28] model the label noise as a function of time, which has a low noise level at the beginning and end stage and a high noise level in the middle. However, this function does not distinguish between the label flipping from positive to negative and negative to positive. Huang et al. [29], in an attempt to learn a noisy-label-robust distance metric, statistically produce all potentially true class labels from the observed labels using a preset label uncertainty parameter, giving rise to a combinatorial optimization problem which is difficult to solve. Our method is different from previous methods in that: (1) each data point has multiple labels from different annotators; (2) each annotator, due to subjectivity and varied expertise, tends to vary in labeling accuracy; (3) we treat the true label of each data point as latent variables. Our method is structurally similar to [30] in that the label annotation task is crowdsourced to a number of different annotators. However, [30] relates each data point to the label likelihood via a set of logistic regression weights, whereas we relate each data point to the label likelihood via a distance metric matrix.

3. Problem setting and overview of our system

In this paper, we focus on the recognition of subtle facial expressions that are spontaneously produced. To this end, the Moving Faces and People (MFP) dataset [32] is used in our work. Here we briefly introduce MFP and describe how we adapt this dataset to suit our needs.

MFP is a large-scale database of static images and video clips of human faces and people. The major difference between this dataset and popular posed expression datasets (e.g. CK+ [31], MMI [33], JAFFE [34]) is that it is collected in a non-intrusive manner in order to facilitate image/video understanding in the natural environment. Part of the whole dataset, the Dynamic Facial Expressions recordings, contains spontaneous expressions of subjects that are induced by a 10-min video with scenes from various movies and television programs. MFP has 10 different expression categories. In this work, only the 7 basic facial expressions (including neutral) as defined in [8] are considered. Fig. 1 shows the comparison between posed and spontaneous disgust, taken from CK+ and MFP respectively.

We wish to note several characteristics of the spontaneous facial expressions in MFP. First, subjects do not necessarily respond with the intended facial expression. For example, a subject may show surprise instead of disgust when watching a disgust-inducing video. Second, expressions vary significantly in length. Some occur over a few frames, others last many seconds. Third, some expressions mix with each other, e.g. faces may change back and forth between fear and disgust. Finally, these recordings are only marked by the expression inducing

Fig. 1. A comparison between posed and spontaneous expressions. (a) Posed disgust, CK+ [31]. (b) Spontaneous disgust, MFP [32].

Table 1
Statistics of the validated MFP dataset. The number of examples under each category is obtained from the gold standard labels. The mislabel ratio for each category is calculated as the ratio of the number of erroneously assigned labels to the total number of labels.

Expression	Neutral	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Number of examples	374	79	132	65	164	89	108
Mislabel ratio	0.0	0.28	0.08	0.18	0.0	0.04	0.0

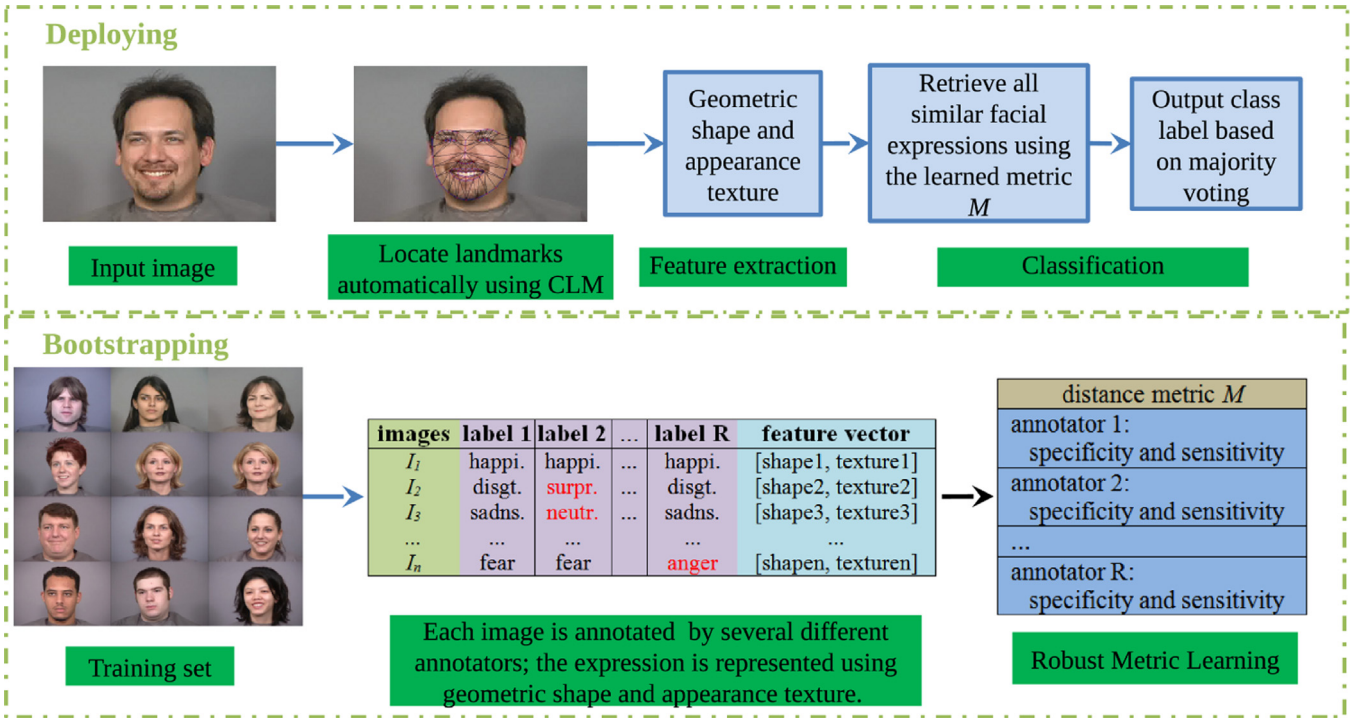

Fig. 2. Flowchart of the facial expression recognition system (best viewed in color). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

video type. *Neither landmarks nor expression labels for the frames of these recordings are available.*

To suit our needs, we validate this dataset by first extracting individual frames from the recordings. Then, the per-frame expression labeling tasks are assigned to annotators. Specifically, 13 annotators who are familiar with facial expression analysis as well as the experiment's design and purpose are involved in the task. As a final step, each face image is registered with 68 landmark points to represent the facial shape.

Apart from the noisy labels obtained from the multiple annotators, a single gold standard label is assigned to each expression based on majority voting, that is, the one with more than half of the votes from the 13 annotators. In case no majority consent is reached, votes from "experts" are solicited, and the gold standard label is determined as the one with the most votes (the vote from an "expert" is weighted twice more than that of an average annotator). The gold standard labels will be used for evaluating the performance of the classifiers.

The statistics of the validated dataset is summarized in Table 1. The number of examples under each category is based on the gold standard labels. The mislabel ratio is the ratio between the number of mislabels and the number of examples of that category. As indicated, some expressions, e.g. fear and disgust, tend to admit a high mislabel ratio due to the insufficient agreement among

annotators, while some other expressions, e.g. happiness and surprise, achieve unanimity among annotators. We will see later in the experiment that our method outperforms the state-of-the-art by making much fewer mistakes in recognizing these easy-to-mislabel expressions.

Our spontaneous facial expression recognition system (Fig. 2) consists of two processing lines: bootstrapping and deploying. In bootstrapping, the distance metric that is specific to spontaneous facial expression recognition is learned from noisy multiple annotations. In deploying, the landmarks are first located on the input image. Then, a feature vector is built and the final recognition result is obtained based on metric learning based kNN voting. The landmark annotation runs in two modes: manual and automatic. In the manual mode, the landmark points are manually provided; in the automatic mode, the landmark points are located by a Constrained Local Model (CLM) based automatic tool as described in [35].

4. Facial feature extraction

We use a fusion of face shape and texture to represent a facial expression, as shown in Fig. 4. This hybrid representation is able to incorporate local pixel intensity variation pattern while still adhering to shape constraint at a global level.

4.1. Shape feature

The geometric face shape information is captured by a set of 68 points known as landmarks. To remove variations in scale, orientation, and reference point, Procrutes Analysis is employed to align these shapes to the mean shape. Fig. 3 illustrates the superimposition of 7 basic expressions from the validated MFP before and after Procrustes alignment.

4.2. Texture feature

The Gabor filters, with kernels similar to the 2D receptive field profiles of the mammalian cortical simple cells [36], have been reported to give improved facial expression recognition performance [18,19]. Therefore, we use as facial texture features the Gabor features.

To have face images with normalized shape and intensity, we first linearly warp the image so that the face shape in the resulting image is aligned with the mean shape. Then, the self-quotient image is calculated to attenuate illumination variation, which is obtained via a per-pixel division operation between the original image and its Gaussian smoothed version.

Our Gabor filter bank consists of filters at 5 scales and 8 orientations. Studies in psychology show that facial features of expressions are located around mouth, nose, and eyes, and their locations are essential for explaining and categorizing facial expressions [37]. Therefore, 7 local patches located around corresponding landmark points are chosen as the expression salency regions, as shown in Fig. 4. Gabor features are then calculated from these 7 patches respectively, resulting in a feature vector of dimension 560. Principal Component Analysis (PCA) is employed to reduce data dimension to 80 while retaining 98% of energy. Denoting the face shape vector and the Gabor feature vector as s and g respectively, a particular expression could be represented as

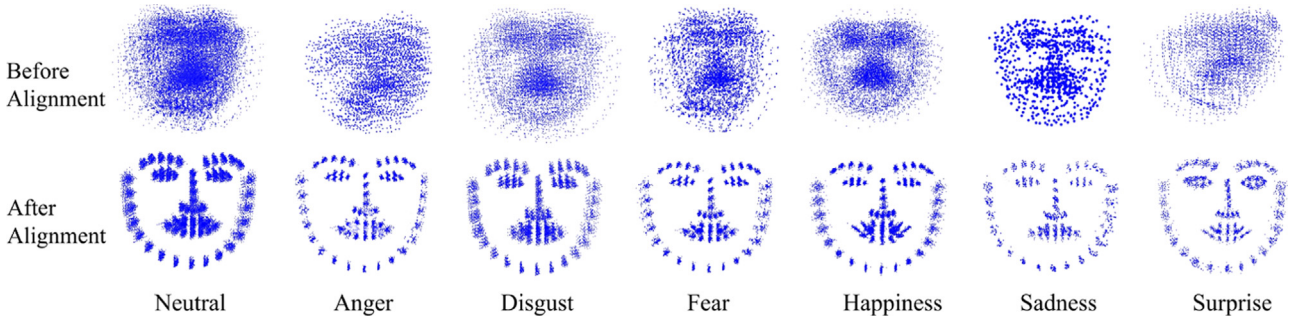

Fig. 3. The superimposition of the shapes of 7 basic expressions from the validated MFP before and after Procrustes alignment.

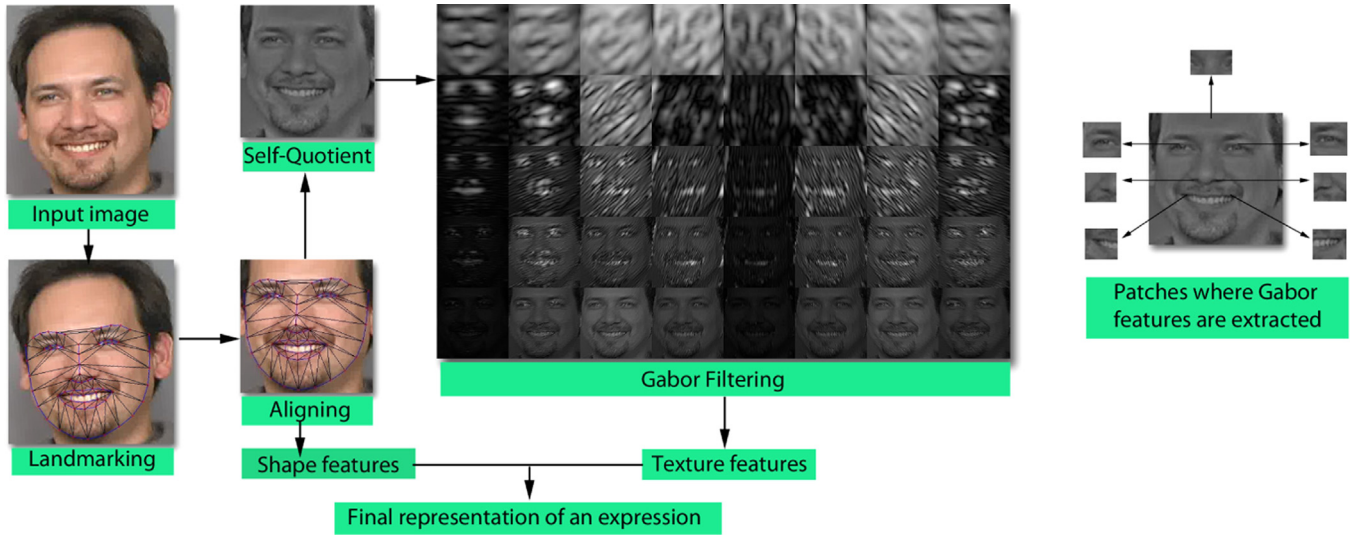

Fig. 4. Feature extraction for a facial expression.

a concatenation s of g:

$$x = [s^T, \mu \cdot g^T]^T \tag{2}$$

where μ is a weighting factor balancing the relative importance of shape and texture. To further reduce data dimension, PCA is performed on x to derive the final representation of facial expression. Without causing confusion, we will still use x to represent facial expression in the later parts of this paper.

To select a proper μ such that s and g are commensurate, we estimate the effect of varying s on g using a similar method in [38]. To do this, we displace s from its ground truth position and the RMS change in g per unit RMS change in s is recorded. The weighting factor μ is set as the inverse of the average value of RMS change of all training examples.

5. Robust metric learning for spontaneous facial expressions

In this section, we first formulate the problem of spontaneous facial expression recognition using a Robust Metric Learning approach, then give an efficient solution to the resulting optimization problem via Expectation Maximization.

5.1. Robust metric learning

Let $\mathcal{D} = \{(x_i^1, x_i^2, t_i^1, t_i^2, \ldots, t_i^R)\}_{i=1}^n$ denote a training set of n labeled inputs, where (x_i^1, x_i^2) is the ith pair of facial expressions, t_i^j is a binary label which evaluates to "1" if (x_i^1, x_i^2) is considered to be in the same class by the jth annotator and "0" otherwise, R is the number of annotators.

Ideally, two facial expressions have a high (low) probability of being in the same (different) class if they have a short (great) distance between them. However, this is not necessarily the case in the Euclidean space due to the overlapping of geometric and appearance features. To address this problem, we propose to learn a discriminative feature space that respects the semantic categories of the data. Formally, given a pair of facial expressions (x_i^1, x_i^2), the probability of them being in the same class can be written as

$$p_i = \Pr(\hat{t}_i = 1 | x_i^1, x_i^2; M) = \sigma(b - d_M(x_i^1, x_i^2)) \tag{3}$$

where $\hat{t}_i$ is the gold standard label for the ith pair of expressions, $d_M(x_i^1, x_i^2)$ is the generalized Mahalanobis distance as defined in Eq. (1), b is the threshold separating expressions of different classes and is set to 1 in this work, $\sigma(z) = (1+\exp(-z))^{-1}$ is the sigmoid function, which without any prior information makes the optimal choice in approximating the conditional distribution of the label given the distance between a pair of expressions [39].

In traditional supervised classification, $\hat{t}_i$ is assumed to be the same as the observed class label, and the optimal distance metric can be obtained by maximizing the likelihood of the labels

$$\max_{M \succeq 0} \sum_{i=1}^{n} \hat{t}_i \ln p_i + (1-\hat{t}_i) \ln(1-p_i) \tag{4}$$

where $M \succeq 0$ restricts M to be a positive semidefinite matrix. Unfortunately, the observed labels tend to contain errors, and directly using them for training usually biases the classifier and results in degraded recognition performance. Motivated by [30], we use sensitivity and specificity to model the fact that there contains mislabeled expressions in the training set and different annotators vary in their annotation reliability. In particular, we define

$$\text{sensitivity}: \ \alpha = [\alpha_1, \alpha_2, \ldots, \alpha_R]$$
$$\text{specificity}: \ \beta = [\beta_1, \beta_2, \ldots, \beta_R]$$

where $\alpha_j = \Pr[t^j = 1 | \hat{t} = 1]$ is the probability that the jth annotator assigns "1" when the true label is "1", and $\beta_j = \Pr[t^j = 0 | \hat{t} = 0]$ is the probability that the jth annotator assigns "0" when the true label is "0". Clearly, the following probability holds

$$\Pr[t^j = 0 | \hat{t} = 1] = 1 - \alpha_j$$
$$\Pr[t^j = 1 | \hat{t} = 0] = 1 - \beta_j$$

Hence, the probability of the jth annotator labeling (x_i^1, x_i^2) as t_i^j can be written as

$$\Pr[t_i^j | x_i^1, x_i^2; M, \alpha_j, \beta_j] = t_i^j[\alpha_j p_i + (1-\beta_j)(1-p_i)] + (1-t_i^j)[(1-\alpha_j)p_i + \beta_j(1-p_i)] \tag{5}$$

Assuming that the annotators assign labels independently, the negative log-likelihood of the labels given the expression pairs can be written as

$$\mathcal{L}(\mathcal{D}|M, \alpha, \beta) = -\sum_{i=1}^{n}\sum_{j=1}^{R} \{t_i^j \ln[\alpha_j p_i + (1-\beta_j)(1-p_i)] + (1-t_i^j) \ln[(1-\alpha_j)p_i + \beta_j(1-p_i)]\} \tag{6}$$

In the following, we will use $\mathcal{L}(M, \alpha, \beta)$ as an equivalent notation of $\mathcal{L}(\mathcal{D}|M, \alpha, \beta)$. The optimal model parameters can be found as

$$(M^*, \alpha^*, \beta^*) = \arg\min_{M, \alpha, \beta} \mathcal{L}(M, \alpha, \beta) \tag{7}$$

Algorithm 1. Expectation Maximization.

Inputs:
Training set $\mathcal{D} = \{(x_i^1, x_i^2, t_i^1, \ldots, t_i^R)\}_{i=1}^n$
Initialize:
$\forall i = 1, \ldots, n$, initialize the true label of the ith pair (x_i^1, x_i^2) using majority voting

$$\hat{t}_i \leftarrow \begin{cases} 1 & \text{if } \left(\sum_{j=1}^{R} 1_{t_i^j = 1}\right) \Big/ \left(\sum_{j=1}^{R} 1\right) \geq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

Repeat:
M-step:
$\forall j = 1, \ldots, R$, estimate the sensitivity for the jth annotator by

$$\alpha_j \leftarrow \left(\sum_{i=1}^{n} 1_{t_i^j = 1, \hat{t}_i = 1}\right) \Big/ \left(\sum_{i=1}^{n} 1_{\hat{t}_i = 1}\right),$$

$\forall j = 1, \ldots, R$, estimate the sensitivity for the jth annotator by

$$\beta_j \leftarrow \left(\sum_{i=1}^{n} 1_{t_i^j = 0, \hat{t}_i = 0}\right) \Big/ \left(\sum_{i=1}^{n} 1_{\hat{t}_i = 0}\right),$$

estimate the metric matrix by

$$M \leftarrow \arg\min_{M \succeq 0} \mathcal{L}(M, \alpha, \beta)$$

E-step:
$\forall i = 1, \ldots, n$, predict the true label of the ith pair (x_i^1, x_i^2) by

$$\hat{t}_i \leftarrow \begin{cases} 1 & \text{if } p_i \geq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

Until convergence
Outputs:
M^*, α^*, β^*

5.2. Parameter estimation

5.2.1. The Expectation Maximization framework

Intuitively, the sensitivity α, the specificity β and the distance metric M can be estimated only when the true class labels are known, and vice visa. To address this chicken-and-egg problem, we present an Expectation Maximization (EM) based algorithm that jointly estimates the annotator sensitivity/specificity, the

distance metric and the actual true labels. The EM algorithm for solving Eq. (7) proceeds by iterating between two main steps, the E-step and the M-step. The E-step estimates the true label of each pair of expressions based on the label likelihood calculated using the current metric M. The M-step updates (M,α,β) using the estimated true labels. See Algorithm 1 for details.

In Algorithm 1, 1_A is defined to be an indicator function that evaluates to "1" if the Boolean expression A is true and "0" otherwise. Note that at line 12, one has to solve

$$\arg\min_{M\succeq 0}\mathcal{L}(M,\alpha,\beta) \tag{8}$$

of which the major computational challenge arises from the positive semidefinite (p.s.d.) constraint $M\succeq 0$.

Existing methods for dealing with p.s.d. constrained optimization problem fall into two categories. The first category of methods derives an equivalent form of the original problem where the p.s.d. constraint is replaced by other easy-to-handle constraints [40,41]. Our method, *Adaptive Online Metric Learning*, belongs to the second category, which directly solves for the optimal p.s.d. metric matrix via gradient descent.

5.3. Adaptive online metric learning

Let k denote the iteration index when solving Eq. (8). A typical batch gradient descent method updates M^k to M^{k+1} by first taking a step along the steepest descent direction of all data points and then projecting back to the cone of p.s.d. matrices,[2] i.e.

$$M^{k+1}\leftarrow\pi_{\mathcal{S}_+}(M^k-\lambda\nabla_M\mathcal{L}(M^k,\alpha,\beta)) \tag{9}$$

where $\pi_{\mathcal{S}_+}(\cdot)$ denotes the projection onto the cone of p.s.d. matrices. Since batch gradient descent requires expensive evaluation of the gradients from all data points, it scales poorly given a large scale dataset.

Motivated by the recent advances in online learning [24], we propose an Adaptive Online Metric Learning algorithm (Algorithm 2) for efficiently solving Eq. (8). In contrast to the batch gradient descent method, Algorithm 2 takes the steepest descent direction, $\nabla_M\mathcal{L}_k(M^k,\alpha,\beta)$, that diminishes the loss incurred only by the current single data point (x_k^1,x_k^2). The iteration stops after all the n data points are processed one by one. Iterative methods of this form are known to converge to the same solution as batch gradient descent, provided that the gradient descent step size is sufficiently small [42].

Also note that instead of using a fixed gradient descent step size, we use an adaptively adjusted one that is inversely proportional to $\sum_{j=1}^{R}t_k^j p_k+(1-t_k^j)(1-p_k)$. The intuition is that, if p_k, the probability of the true label being "1", is highly consistent with the annotated labels, then M only needs to be updated with a small step. Therefore, the step size is set to be $\lambda/(\sum_{j=1}^{R}t_k^j p_k+(1-t_k^j)(1-p_k))$.

Algorithm 2. Adaptive Online Learning for Eq. (8).

1: **Inputs**:
   $\mathcal{D}=\{(x_i^1,x_i^2,t_i^1,\ldots,t_i^R)\}_{i=1}^n$,
   α,β: the sensitivity and specificity calculated in the current M-step,
   M^1: the metric at which the previous M-step converges,
   λ: the step size.
2: **for** $k=1\rightarrow n$ **do**
3: receive $(x_k^1,x_k^2,t_k^1,\ldots,t_k^R)$ and update M^k to M^{k+1} by
4: $M^{k+1}\leftarrow\pi_{\mathcal{S}_+}(M^k-\frac{\lambda}{\sum_{j=1}^{R}t_k^j p_k+(1-t_k^j)(1-p_k)}\nabla_M\mathcal{L}_k(M^k,\alpha,\beta))$
5: **end for**
6: **Outputs**:
   M

Next, we show how to perform the p.s.d. cone projection $\pi_{\mathcal{S}_+(\cdot)}$ in an efficient way. Let

$$\eta=\frac{\lambda\left\{\frac{t_k^j[\alpha_j+(1-\beta_j)]}{\alpha_j p_k+(1-\beta_j)(1-p_k)}+\frac{(1-t_k^j)[(1-\alpha_j)-\beta_j]}{(1-\alpha_j)p_k+\beta_j(1-p_k)}\right\}p_k(1-p_k)}{\sum_{j=1}^{R}t_k^j p_k+(1-t_k^j)(1-p_k)} \tag{10a}$$

$$\eta'=(x_k^1-x_k^2)^T(M^k)^{-1}(x_k^1-x_k^2) \tag{10b}$$

It is easy to show that

$$M^k-\frac{\lambda}{\sum_{j=1}^{R}t_k^j p_k+(1-t_k^j)(1-p_k)}\nabla_M\mathcal{L}_k(M^k,\alpha,\beta)=M^k-\eta(x_k^1-x_k^2)(x_k^1-x_k^2)^T \tag{11}$$

Then, instead of using eigen-decomposition to perform the p.s.d. projection of Eq. (11), we use the following formula

$$\pi_{\mathcal{S}_+}(M^k-\eta(x_k^1-x_k^2)(x_k^1-x_k^2)^T)=\begin{cases}M^k-\eta(x_k^1-x_k^2)(x_k^1-x_k^2)^T & \text{if } \eta\le\eta'\\ M^k-\eta'(x_k^1-x_k^2)(x_k^1-x_k^2)^T & \text{otherwise}\end{cases} \tag{12}$$

Proof (*Derivation*). See Appendix A.

In summary, our Adaptive Online Metric Learning algorithm not only achieves faster convergence via adaptively adjusting the gradient descent step size, but also significantly reduces the computational workload by replacing the eigen-decomposition based p.s.d. cone projection with a simple two-case function. Overall, a significant efficiency gain is achieved by Algorithm 2.

6. Experiments

After learning the robust distance metric, given a novel facial expression, its class label can be identified by first retrieving the training examples that are predicted to be in the same class as the novel expression, then performing majority voting using the actual expression labels of these training examples. Two groups of experiments are conducted in our study. In the first group, Robust Metric Learning is extensively evaluated against a number of the state-of-the-art methods in terms of spontaneous expression recognition accuracy. In the second group, the spontaneous-specific metric is used to recognize posed expressions via transfer learning to see how well it generalizes.

6.1. Comparison of recognition accuracy

To justify the notion of Robust Metric Learning based spontaneous expression recognition, we conduct extensive comparative experiments using the MFP dataset. In total, 8 different methods are compared, including

1. EUC: the baseline Euclidean distance metric.
2. Isomap: [43], 20 nodes for neighborhood graph construction, followed by SVM classification.
3. LLE: Locally Linear Embedding [44], 20 nodes for neighborhood graph construction, followed by SVM classification.
4. LDA: Linear Discriminant Analysis.

[2] The p.s.d. cone projection amounts to finding the nearest p.s.d. matrix in the sense of least squared difference.

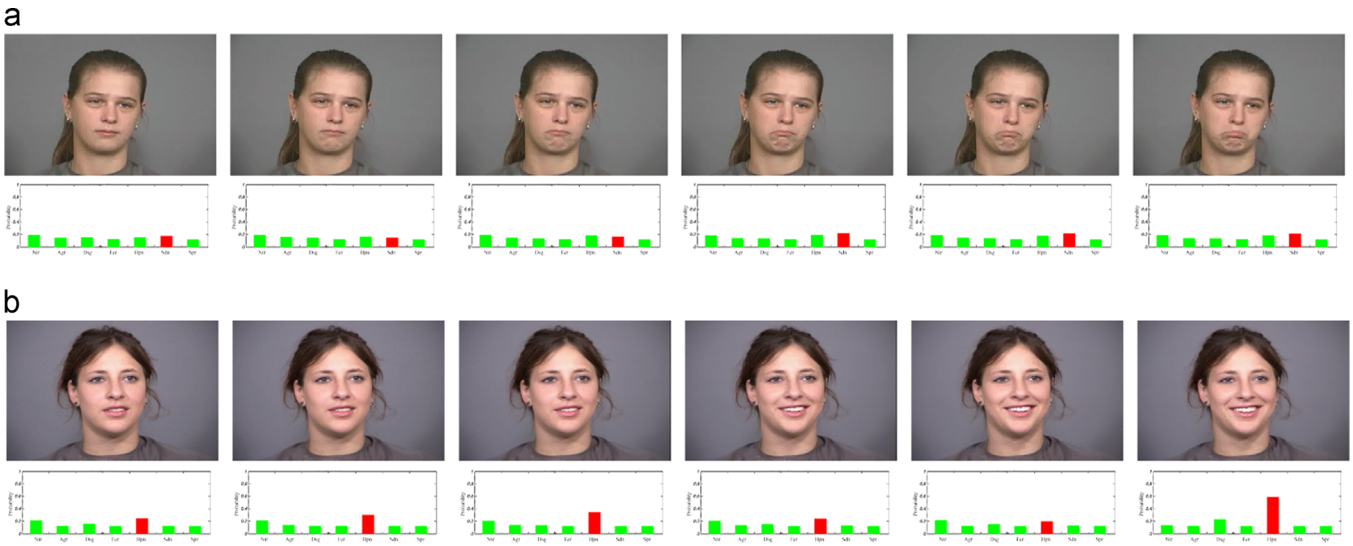

Fig. 5. Recognition examples using Robust Metric Learning. The 7 bars under each face image, from left to right, correspond to neutral, anger, disgust, fear, happiness, sadness, and surprise respectively. Bar height is the normalized kNN voting. Both sequences start with neutral and end in the peak expression. Happiness, with salient features such as wide-open mouth, can be recognized effortlessly. Sadness is much less distinguishable from other expressions, however, our Robust Metric Learning method still manages to get the correct result via the domain adapted metric. (a) Recognizing sadness with effort. (b) Recognizing happiness with ease.

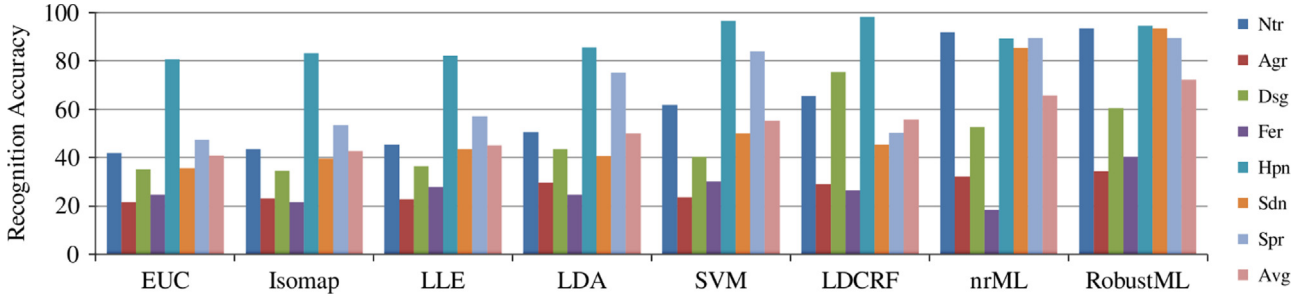

Fig. 6. Comparison between various methods on spontaneous expression recognition.

5. SVM: Support Vector Machine, RBF kernel [45].
6. LDCRF: Latent-Dynamic Conditional Random Field, a temporal model of internal micro-structures of facial expressions [12].
7. RobustML: the Robust Metric Learning method proposed in this paper.
8. nrML: the non-robust version of metric learning.

For all methods but RobustML, only one label per image is used for training, which is simply determined from the noisy labels based on majority voting. The facial landmarks are located automatically using the tool described in [35].[3] We split each class of expressions by a ratio of 3/1 for training/testing. To reduce variability, 10 rounds of validation are performed using randomly generated splits, and the final results are calculated by averaging over all the 10 rounds.

Fig. 5a and b shows two recognition examples using Robust Metric Learning. Happiness, with salient features such as wide-open mouth, can be recognized effortlessly. Sadness is much less distinguishable from other expressions, however, our Robust Metric Learning method still manages to get the correct result via the domain adapted metric.

Fig. 6 shows the per-class recognition rates of different methods.[4] For space limitation, we only give the confusion matrix obtained from SVM and RobustML respectively in Tables 2 and 3. Two observations can be made here. First, different methods vary a great deal in recognizing "hard" expressions such as neutrality, fear and sadness. In particular, EUC is the least accurate. LDCRF, by modeling the temporal variations of face shape and texture, achieves much higher recognition accuracy than other non-metric learning based methods. RobustML and nrML, with the domain-specific distance metric, are the most successful in differentiating the "hard" expressions, thus significantly outperforming all other methods.

Second, RobustML outperforms nrML by 5%. This is because annotators can vary a great deal in their respective annotation reliability. nrML simply discards this important information by assuming equal reliability among all annotators. Fig. 7 shows the estimated (1-sensitivity) and (1-specificity) of the 13 annotators using RobustML, with the annotators sorted by (1-sensitivity). As can be seen, (1-sensitivity) ranges from 0.013 to 0.12, whereas (1-specificity) ranges from 0.008 to 0.072, thus proving the assumption of equally reliable annotators to be invalid. In addition, (1-sensitivity) is consistently greater than (1-specificity). This is because facial expressions of different classes are more likely to be

[3] In the manual mode in which landmarks are manually provided, all methods compared here give significantly better average recognition accuracy. In particular, SVM gives 70.3% as compared to 55.2% in the automatic mode; RobustML gives 77.8% compared to 72.27% in the automatic mode. Due to space limitation, the detailed experimental results from the manual mode are not presented.

[4] The average recognition rate is the mean of per-class recognition rates.

Table 2
The confusion matrix obtained from SVM for recognizing spontaneous facial expressions.

Expressions	Neutral	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Neutral	**61.8**	8.7	4.9	4.4	0.0	20.2	0.0
Anger	20.1	**23.5**	18.9	13.2	3.4	17.3	3.6
Disgust	25.2	10.6	**40.4**	14.2	0.0	6.9	2.7
Fear	17.9	22.9	14.2	**30.2**	4.2	8.9	1.7
Happiness	0.0	1.2	2.1	0.0	**96.5**	0.0	0.2
Sadness	34.2	4.0	5.8	5.9	0.0	**50.1**	0.0
Surprise	0.0	3.1	2.5	4.1	6.2	0.0	**84.1**

Table 3
The confusion matrix obtained from Robust Metric Learning for recognizing spontaneous facial expressions.

Expressions	Neutral	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Neutral	**93.5**	1.2	0.0	1.4	0.0	3.9	0.0
Anger	14.8	**34.2**	20.1	10.5	5.2	7.8	7.4
Disgust	20.4	2.4	**60.4**	6.5	2.2	0.0	8.1
Fear	20.4	19.7	15.3	**40.4**	0.0	4.2	0.0
Happiness	0.0	0.7	4.9	0.0	**94.4**	0.0	0.0
Sadness	6.3	0.2	0.0	0.0	0.0	**93.5**	0.0
Surprise	0.0	1.1	0.0	4.2	5.2	0.0	**89.5**

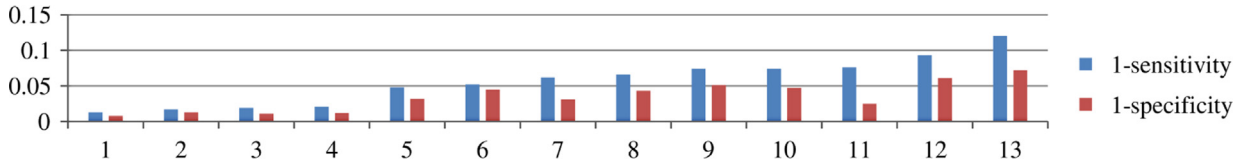

Fig. 7. The (1-sensitivity) and (1-specificity) value of the 13 annotators, sorted by (1-sensitivity).

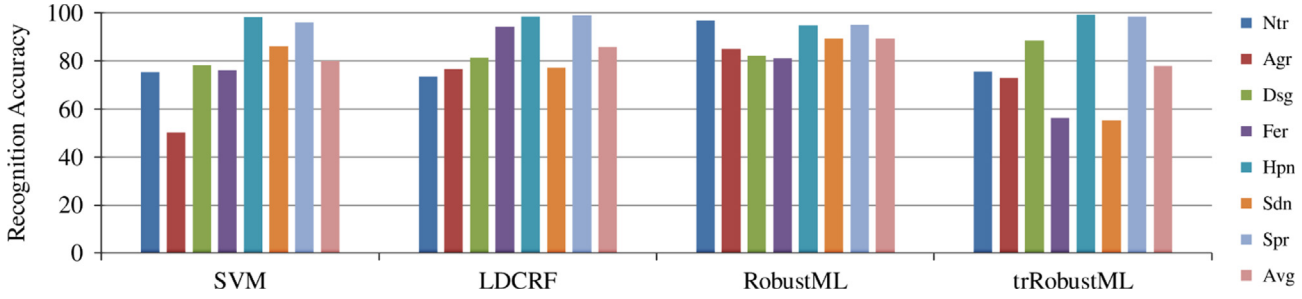

Fig. 8. Generalization of the spontaneous-specific metric to posed expressions.

labeled as the same, but the same facial expressions are much less likely to be labeled as different.

6.2. Generalization of the learned metric

In this experiment, we investigate how well our method generalizes to posed expressions. This is of particular interest to us because the expensive data labeling and retraining can be avoided if the knowledge learned in one domain can be transferred to another. To this end, we first learn a spontaneous-specific distance metric, then apply it to the posed expressions from the CK+ dataset [31].

To enable knowledge transfer, we take the transductive transfer learning approach as described in [46]. In particular, given the labeled source data from MFP and unlabeled target data from CK+, we learn an optimal weighting of the source data so that its distribution resembles the empirical distribution of target data the most. (See [46] for details of this algorithm.) Finally, we use the weighted source data to learn the distance metric, and test it on the target data from CK+. In addition, we also train and test SVM, LDCRF and RobustML directly on the target data. Per-class recognition rate and average recognition rate are given in Fig. 8.

As can be seen, directly trained RobustML (RobustML) is better than transferred RobustML (trRobustML) for recognizing posed expressions. What is noteworthy is that trRobustML, with a recognition accuracy of 78.1%, achieves comparable performance to the directly trained SVM. In summary, our method compensates for knowledge transfer loss by its discriminative and corrective power, thus generalizing well to posed expressions.

7. Conclusion

In this work, we propose a Robust Metric Learning method for spontaneous facial expression recognition. In contrast to traditional supervised classification methods, we explicitly take into account the potential label errors when designing our method. In particular, we collect subjective (possibly erroneous) labels from

multiple annotators. In practice, there is a substantial amount of disagreement among the annotators. The proposed Expectation Maximization based framework iteratively establishes a particular gold standard, measures the performance of the annotators given that gold standard, and then refines the gold standard based on the performance measures. In the meantime, to alleviate the possible overlapping of geometric and appearance features of different facial expressions, we iteratively update the distance metric with the newly estimated annotator performance. The resulting classification procedure is simply a majority voting process based on the training examples retrieved using the learned distance metric.

One drawback of the current model is that it assumes each annotator maintains his/her performance across different expression classes. In practice, the annotator performance depends crucially on the facial expression he/she is labeling (An annotator is less prone to error when labeling easier expressions.) and there is some degree of correlation among the annotators (due to culture background, psychological perception, etc.). A simple extension can be made by making the annotator sensitivity and specificity depend on the class labels. This is subject to future work.

Conflict of interest

None declared.

Appendix A. Derivation of Eq. (12)

Proof (*Derivation*). According to Schur Complement condition,

$$M^k-\eta(x_k^1-x_k^2)(x_k^1-x_k^2)^T\succcurlyeq 0 \tag{A.1}$$

is equivalent to

$$\begin{pmatrix} M^k & x_k^1-x_k^2 \\ (x_k^1-x_k^2)^T & \eta^{-1} \end{pmatrix}\succcurlyeq 0$$

which, again by Schur Complement condition, is equivalent to

$$\eta^{-1}-(x_k^1-x_k^2)^T(M^k)^{-1}(x_k^1-x_k^2)\geq 0 \tag{A.2}$$

or, equivalently

$$\eta\leq((x_k^1-x_k^2)^T(M^k)^{-1}(x_k^1-x_k^2))^{-1} \tag{A.3}$$

Let $\eta'=((x_k^1-x_k^2)^T(M^k)^{-1}(x_k^1-x_k^2))^{-1}$, then the projection to the p.s.d. cone can be efficiently computed as

$$\pi_{\mathcal{S}_+}(M^k-\eta(x_k^1-x_k^2)(x_k^1-x_k^2)^T)=\begin{cases} M^k-\eta(x_k^1-x_k^2)(x_k^1-x_k^2)^T & \text{if } \eta\leq\eta' \\ M^k-\eta'(x_k^1-x_k^2)(x_k^1-x_k^2)^T & \text{otherwise} \end{cases} \tag{A.4}$$

we thus have the result. □

References

[1] I. Kotsia, S. Zafeiriou, I. Pitas, Texture and shape information fusion for facial expression and facial action unit recognition, Pattern Recognit. 41 (2008) 833–851.

[2] M. Kyperountas, A. Tefas, I. Pitas, Salient feature and reliable classifier selection for facial expression classification, Pattern Recognit. 43 (2010) 972–986.

[3] H.-Y. Chen, C.-L. Huang, C.-M. Fu, Hybrid-boost learning for multi-pose face detection and facial expression recognition, Pattern Recognit. 41 (2008) 1173–1185.

[4] M.F. Valstar, M. Pantic, Z. Ambadar, J.F. Cohn, Spontaneous vs. posed facial behavior: automatic analysis of brow actions, in: Proceedings of the 8th International Conference on Multimodal Interfaces, ICMI '06, ACM, New York, NY, USA, 2006, pp. 162–170.

[5] J. Cohn, K. Schmidt, The timing of facial motion in posed and spontaneous smiles, Int. J. Wavel., Multiresolut. Inf. Process. 2 (2004) 1–12.

[6] S. Wan, J.K. Aggarwal, A scalable metric learning-based voting method for expression recognition, in: 10th IEEE International Conference on Automatic Face and Gesture Recognition (FG), 2013, pp. 1–8.

[7] P.J.N.J.A. Russell, Judgments of emotion from spontaneous facial expressions of new guineans, Emotions 7 (2007) 736–744.

[8] P. Ekman, W.V. Friesen, Unmasking the Face: a Guide to Recognizing Emotions from Facial Clues, Prentice-Hall, Oxford, 1975.

[9] X. Sun, L. Rothkrantz, D. Datcu, P. Wiggers, A Bayesian approach to recognise facial expressions using vector flows, in: Proceedings of the International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing, CompSysTech '09, ACM, New York, NY, USA, 2009, pp. 28:1–28:6.

[10] Y. Wang, H. Ai, B. Wu, C. Huang, Real time facial expression recognition with adaboost, in: Proceedings of the 17th International Conference on Pattern Recognition, vol. 3, pp. 926–929.

[11] S. Das, K. Yamada, Article: an HMM based model for prediction of emotional composition of a facial expression using both significant and insignificant action units and associated gender differences, Int. J. Comput. Appl. 45 (2012) 11–18.

[12] S. Jain, C. Hu, J.K. Aggarwal, Facial expression recognition with temporal modeling of shapes, in: 2011 IEEE International Conference on Computer Vision Workshops, pp. 1642–1649.

[13] M. Pantic, L.J.M. Rothkrantz, Automatic analysis of facial expressions: the state of the art, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 1424–1445.

[14] S. Park, D. Kim, Spontaneous facial expression classification with facial motion vectors, in: IEEE International Conference on Automatic Face and Gesture Recognition, pp. 1–6.

[15] Z. Zeng, M. Pantic, G.I. Roisman, T.S. Huang, A survey of affect recognition methods: audio, visual, and spontaneous expressions, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2009) 39–58.

[16] M. Pantic, L.J.M. Rothkrantz, Facial action recognition for facial expression analysis from static face images, IEEE Trans. Syst., Man, Cybern., Part B: Cybern. 34 (2004) 1449–1461.

[17] M. Pantic, M.S. Bartlett, Machine analysis of facial expressions, in: K. Delac, M. Grgic (Eds.), Face Recognition, I-Tech Education and Publishing, Vienna, Austria, 2007, pp. 377–416.

[18] S.M. Lajevardi, M. Lech, Averaged Gabor filter features for facial expression recognition, in: Proceedings of the 2008 Digital Image Computing: Techniques and Applications, DICTA '08, IEEE Computer Society, Washington, DC, USA, 2008, pp. 71–76.

[19] S. Bashyal, G.K. Venayagamoorthy, Recognition of facial expressions using Gabor wavelets and learning vector quantization, Eng. Appl. Artif. Intell. 21 (2008) 1056–1064.

[20] J. Whitehill, C. Omlin, Haar features for FACS AU recognition, in: 7th International Conference on Automatic Face and Gesture Recognition, pp. 5–101.

[21] S.Z. Li, A.K. Jain (Eds.), Facial Expression Analysis: Handbook of Face Recognition, 2nd edition, Springer, 2011.

[22] E. Xing, A. Ng, M. Jordan, S. Russell, Distance metric learning, with application to clustering with side-information, in: Advances in Neural Information Processing Systems, vol. 15, pp. 505–512.

[23] K.Q. Weinberger, L.K. Saul, Distance metric learning for large margin nearest neighbor classification, J. Mach. Learn. Res. 10 (2009) 207–244.

[24] J.V. Davis, B. Kulis, P. Jain, S. Sra, I.S. Dhillon, Information-theoretic metric learning, in: Proceedings of the 24th International Conference on Machine Learning, ACM, New York, NY, USA, 2007, pp. 209–216.

[25] A. Globerson, S. Roweis, Metric learning by collapsing classes, in: Advances in Neural Information Processing Systems vol. 18, MIT Press, Cambridge, MA, 2006, pp. 451–458.

[26] M. Guillaumin, J. Verbeek, C. Schmid, Is that you? Metric learning approaches for face identification, in: International Conference on Computer Vision, pp. 498–505.

[27] M. Opper, O. Winther, Mean field methods for classification with gaussian processes, in: Neural Information Processing Systems, pp. 309–315.

[28] Y. Xue, D.P. Williams, H. Qiu, Classification with imperfect labels for fault prediction, in: Proceedings of the First International Workshop on Data Mining for Service and Maintenance, KDD4Service '11, ACM, New York, NY, USA, 2011, pp. 12–16.

[29] K. Huang, R. Jin, Z. Xu, C.-L. Liu, Robust metric learning by smooth optimization, in: Conference on Uncertainty in Artificial Intelligence, pp. 244–251.

[30] V.C. Raykar, S. Yu, L.H. Zhao, G.H. Valadez, C. Florin, L. Bogoni, L. Moy, Learning from crowds, J. Mach. Learn. Res. 11 (2010) 1297–1322.

[31] P. Lucey, J. Cohn, T. Kanade, J. Saragih, Z. Ambadar, I. Matthews, The extended Cohn–Kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 94–101.

[32] A. O'Toole, J. Harms, S. Snow, D. Hurst, M. Pappas, J. Ayyad, H. Abdi, A video database of moving faces and people, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2005) 812–816.

[33] M. Pantic, M.F. Valstar, R. Rademaker, L. Maat, Web-based database for facial expression analysis, in: Proceedings of IEEE International Conference on Multimedia and Expo, Amsterdam, The Netherlands, pp. 317–321.

[34] M. Lyons, S. Akamatsu, M. Kamachi, J. Gyoba, Coding facial expressions with Gabor wavelets, in: Proceedings of the 3rd International Conference on Face and Gesture Recognition, FG '98, IEEE Computer Society, Washington, DC, USA, 1998, pp. 200–207.

[35] J.M. Saragih, S. Lucey, J. Cohn, Face alignment through subspace constrained mean-shifts, in: International Conference of Computer Vision (ICCV).

[36] J.G. Daugman, Complete discrete 2D Gabor transform by neural networks for image analysis and compression, IEEE T. Acoust. Speech. 36 (1988) 1169–1179.
[37] H. Miyata, R. Nishimura, K. Okanoya, N. Kawai, The mysterious Noh mask: contribution of multiple facial parts to the recognition of emotional expressions, PLoS ONE 7 (2012) e50280.
[38] T.F. Cootes, G.J. Edwards, C.J. Taylor, Active appearance models, in: IEEE Trans. Pattern Anal. Mach. Intell., Springer, 1998, pp. 484–498.
[39] J. Mount, The equivalence of logistic regression and maximum entropy models, 2011.
[40] C. Shen, J. Kim, L. Wang, A. van den Hengel, Positive semidefinite metric learning with boosting, in: NIPS, pp. 1651–1659.
[41] J.V. Davis, I.S. Dhillon, Structured metric learning for high dimensional problems, in: Proceedings of the 14th ACM SIGKDD, ACM, New York, NY, USA, 2008, pp. 195–203.
[42] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, New York, NY, USA, 2004.
[43] J.B. Tenenbaum, V.D. Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[44] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.
[45] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (2011) 27:1–27:27.
[46] M. Sugiyama, S. Nakajima, H. Kashima, P.V. Buenau, M. Kawanabe, Direct importance estimation with model selection and its application to covariate shift adaptation, in: Advances in Neural Information Processing Systems, MIT Press, Cambridge, MA, 2008, pp. 1433–1440.

Shaohua Wan received the Bachelor of Science Degree from Beijing University of Posts and Telecommunications in 2011. He entered the University of Texas at Austin in Fall 2011. His research interests include facial expression analysis, similarity based search, metric learning.

J.K. Aggarwal is on the faculty of The University of Texas at Austin College of Engineering and is currently a Cullen Professor of Electrical and Computer Engineering and Director of the Computer and Vision Research Center. His research interests include computer vision, pattern recognition focusing on human motion.