The following publication M. X. Huang, J. Li, G. Ngai, H. V. Leong and K. A. Hua, "Fast-PADMA: Rapidly Adapting Facial Affect Model From Similar Individuals," in IEEE Transactions on Multimedia, vol. 20, no. 7, pp. 1901-1915, July 2018 is available at https://doi.org/10.1109/TMM.2017.2775206

Fast-PADMA: Rapidly Adapting Facial Affect Model from Similar Individuals

Michael Xuelin Huang, Jiajia Li, Grace Ngai, Hong Va Leong, *Member, IEEE Computer Society*, Kien A. Hua, *Fellow, IEEE*

***Abstract*— Conventional supervised techniques for automated facial affect recognition rely on the availability of diverse, well-annotated training data to build models that can accommodate inter-personal differences. However, this requires enormous manual annotation efforts, which involve expert labor and are expensive, and error-prone. This is a bottleneck that limits the use of facial affect recognition in daily applications. User-specific or personalized models alleviate some of the challenges caused by identity bias, but these approaches are similarly constrained by the amount of well-annotated individual data, which is time-consuming and costly to collect.**

This paper proposes a novel user-adaptive model, which we have termed fast-PADMA (fast Personal Affect Detection with Minimal Annotation). Fast-PADMA integrates data from multiple source subjects with a small amount of data from the target subject. Collecting this target subject data is feasible since fast-PADMA requires only one self-reported affect annotation per facial video segment. To alleviate overfitting in this context of limited individual training data, we propose an efficient bootstrapping technique, which strengthens the contribution of *multiple* similar source subjects. Specifically, we employ an ensemble classifier to construct pre-trained weak generic classifiers from data of multiple source subjects, which is weighted according to available data from the target user. The result is a model that does not require expensive computation, such as distribution dissimilarity calculation or model retraining. We evaluate our method with in-depth experimental evaluations on four publicly available facial datasets, with results that compare favorably with state-of-the-art performance on classifying pain, arousal and valence. Our findings show that fast-PADMA is effective at rapidly constructing a user-adaptive model that outperforms both its generic and user-specific counterparts. This efficient technique has the potential to significantly improve facial affect recognition in real-use cases and therefore enable comprehensive affect-aware applications.

***Index Terms*—Affective computing, facial affect, rapid modeling, user-adaptive model.**

This paragraph of the first footnote will contain the date on which you submitted your paper for review. It will also contain support information, including sponsor and financial support acknowledgment. For example, "This work was supported in part by the U.S. Department of Commerce under Grant BS123456".

Michael Xuelin Huang, Jiajia Li, Grace Ngai, and Hong Va Leong are with the Department of Computing, Hong Kong Polytechnic University, Hong Kong (e-mail: {csxhuang, csjjli, csgngai, cshleong}@comp.polyu.edu.hk).

Kien A. Hua is with the School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816 (e-mail: kienhua@eecs.ucf.edu).

I. Introduction

AUTOMATED spontaneous facial affect recognition enables machines to be aware of users' mental states and facilitates potential advancements in human computer interaction. It has been named as a promising technique in recent affective studies [1]. However, the affect model relying on one *generic classifier* ([2][3]) learnt on the *source subjects*' data in the training set has acute problems when it comes to accommodating differences between individuals. This issue of model generalizability constrains the widespread application of facial affect recognition in real-use situations. Theoretically speaking, given sufficient data, a *user-specific model* (trained only on data from the *target* user) can achieve ideal recognition accuracy, for it can be well customized for the *target* user, accounting for the facial geometry and personal expression.

One challenge of user-specific modeling is acquisition of annotated data. Given a video segment of a user's facial expression, conventional methods require that every frame be annotated with the user's affect at that moment [1]. Clearly, this manual annotation is expensive, tedious, error-prone, and rarely feasible in real-use situations where large amounts of data are needed for robust performance. Fortunately, most real world applications are more concerned about the *overall* human affect over a period, such as the level of engagement while reading an article, or the level of interest while watching an advertisement clip, rather than the momentary, frame-level affect. This makes multiple-instance learning (MIL) attractive. MIL mainly learns and classifies at the bag level. In the case of facial affect learning, each bag is a video segment which contains multiple frames/instances, and only the segment-level annotation is required for training. This means that frame-level annotation is no longer necessary, thus the annotation effort can be drastically reduced.

Although MIL reduces the annotation effort, in most cases, the available target data, or data from the target user, is usually limited compared to the source data, which can be pre-collected. Effective usage of this target data is therefore critical. There have been some attempts at using transfer learning for facial affect recognition with limited target data [1]. These techniques usually adopt the instance-transfer approach [3], which identifies instances that are similar to those of the target, and re-weighs and heightens the importance of those instances.

However, it is not wise to down-weigh all dissimilar instances as some dissimilar instances are necessary if a model is to be able to handle unseen data. Furthermore, instance-transfer is computationally expensive. Deploying the model also requires that the source data instances be available on the client side, which may not be desirable or even possible in certain contexts.

To accelerate the construction of the target classifier, some efforts have focused on model personalization from multiple individual classifiers, each of them pre-trained on annotated data from one source subject [2][4]. However, there are three drawbacks. First, the amount of well-annotated data for each source subject is likely to be small, especially in facial affect recognition contexts involving spontaneous expressions, where obtaining the frame-level annotations mean laborious labeling. As a result, the individual classifiers, which are the foundation of the knowledge transfer, are likely to suffer from overfitting. Second, if knowledge adaptation only considers the distribution of the target data instances in the feature space, and does not consider the semantic meaning, or the label, behind the instances, it can be susceptible to user differences, which is also the problem commonly suffered by generic classifiers. Third, to allow for rapid parameter estimation based on distribution of data [2][4], each classifier needs to use the same mapping function. This limitation constrains the modeling capability of an otherwise sophisticated system.

This paper proposes a novel and efficient approach to building a user-adaptive facial affect model, which bridges the gap between the limited individual data and a practical, well-performing affect model. We assume that there is a certain degree of commonality between the target user and the source subjects *in general*. In other words, the target user shares certain individual characteristics with *some* of the source subjects. However, instead of building an ensemble of multiple *individual* classifiers, our approach is a radically different strategy that starts with *weak generic classifiers*, each of which is trained on data from a subset of *multiple* source subjects, and identifies groups of individuals who are similar to the target subject and accentuates their importance. To reduce the personal geometric bias in the weak generic classifier, we align the frame-level feature vectors into a new feature space, considering both the individuals' expressionless state and their feature boundaries. We also propose a new feature representation to facilitate the knowledge transfer at the segment level.

The contributions of this paper are as follows. We (1) propose an efficient bootstrapping-based technique to transfer the generic knowledge of facial affects; (2) devise an alignment technique to normalize data across diverse individuals for the weak generic classifiers; (3) develop a simple but effective method to aggregate the segment-level feature for multiple-instance learning and (4) present state-of-the-art facial affect classification performances on four public datasets. In contrast to previous studies, our method rapidly builds an adaptive model for the target user, without storing the training instances or depending on computational optimization. We shall empirically show that it can effectively adapt to user individuality and outperform the generic and user-specific counterparts.

The rest of this paper is organized as follows. In Section II we provide the summary of related work. Our system overview is presented in Section III. Section IV introduces the proposed techniques. The experiment setup and result are presented in Section V. Finally, we conclude this paper and discuss our future work in Section VI.

II. Related Work

The goal of this paper is to rapidly build a user-adaptive facial affect model. Two pertinent essential issues are the utilization of the segment-level annotation and the adaptation of the source subjects' data. This section reviews previous work addressing these two issues.

A. Learning Facial Affect from Bag Annotation

Multiple-instance learning refers to machine learning approaches in which a set, or "bag", of instances shares a common overall label, or a "bag annotation". In the context of facial affect learning from video, an instance is a frame from a video segment, a bag is the segment itself, and a label is the user affect. Recent studies generally follow three main approaches when learning from the segment annotation. The first approach assigns all the instances in a bag with the bag label. Viola et al. [5] developed a boosting variant called MILBoost, which initializes all the instances (e.g. individual frames) with the label of the bag and applies boosting for further learning. Sikka et al. [6] extracted facial gestures from a video segment and employed MILBoost for pain recognition from facial expression. These methods assume that a large proportion of the instances in a bag coincide with the annotated label of the bag. However, this assumption may not hold in real-use facial affect recognition systems, as it is not uncommon to have multiple affects occur within a given segment of time in natural contexts.

Rather than use all the instances in a bag, the second approach adopts a subset of them as representation. For instance, Ashraf et al. [7] proposed to cluster the facial expressions in each segment and use the centroids to represent the segment for pain detection. However, the same problem occurs: when mixed emotions are present, and with some emotions that are more momentary in nature (e.g. surprise), the affects exhibited by some centroids may not be consistent with the segment annotation.

The third approach devises a new feature space to characterize a bag. Chen et al. [8] determined bag similarity based on bag-to-instance distances. All instances are used to form the bag-level feature vector. However, this generates a high-dimensional space. Fu et al. [9] simplified the prototype vector by selecting only one instance per bag, which generates a vector with much lower dimension. Xiao et al. [10] explicitly measured the bag dissimilarity taking into consideration the instance similarity between the positive and negative bags. However, since the bag similarity is defined as the pairwise distance between the instances, the computation exponentially increases as the number of instances and bags. Cheplygina et al. [11] studied different forms of prototypes to measure the bag dissimilarity, including representations at instance-level and

bag-level. They then proposed a balanced method using random subspace as the prototype. Despite their success, using a few selected instance(s) for bag description may not be suitable for facial affect recognition, especially for lengthy segments with a diversity of instances, which may not all be representative of the affect depicted by the bag label.

Other efforts focus on facial affect recognition in the MIL paradigm. Ruiz et al. [12] proposed to identify multiple prototypes through an optimization mechanism, which jointly learns the prototypes and the parameters for the bag classifier. However, their study has not suggested the way to determine the number of prototypes. Additionally, since the prototypes and the classifier are jointly determined by the training set, this method may require a computationally expensive optimization for each model update with newly collected data. To reduce the computational cost, Huang et al. [13] encoded the segment characteristic by the probabilities of different emotional frames, but the identification of these emotional frames is still highly constrained by the proportion of positive instances in the training bags. Huang et al. [14] proposed a fast, association-based multiple-instance learning (AMIL) technique to ascertain the indication of facial gestures from their distributions across segments with different annotations. This method can effectively extract useful information from the user-specific data; however, it does not attempt to explore the generic knowledge that might be obtained by considering the similarity between subjects.

In this paper, we exploit AMIL to represent the user-specific facial affect implication. In addition, we introduce statistical pooling to extract the generic attributes of dispersion and distribution of the bag instances. This feature representation is effective for the proposed user-adaptive model.

B. Adapting from Generic Source Data

Much current research in affective computing focuses on model generalization for new users [15]. However, generic, or user-independent models have difficulty accommodating individual differences. Littlewort et al. [16] reported an accuracy drop from 95% to 60% when a model trained on one dataset is tested on another. Michel et al. [17] carried out similar experiments and the accuracy drops from 87.5% to 60.7%. Findings from the first facial expression recognition and analysis (FERA) challenge [18] also show that the user-specific model generally outperforms the user-independent model.

There have been efforts in combining source and target data into the same model. Valstar et al. [18] showed that high performance could be achieved for emotion recognition when prior training data for the target user is available. However, obtaining enough well-labeled user-specific data is expensive for real-use systems. There is also a data skew issue if the data is aggregated directly: the contribution of the target data is likely to be overwhelmed by the much larger source data.

In spite of the previous success of automated facial affect studies in lab scenarios, recognition of spontaneous expression in natural contexts is still challenging [14]. An ideal solution to distinguish subtle expression differences under inter-personal variations is to personalize a user-specific model. Transfer learning has recently become popular for knowledge adaptation and addressing the target data scarcity issue in different problems [21], including document, sentiment, and image classification. For example, Dai et al. [22] extended Adaboost [23] for inductive transfer learning, which assumes some annotated target data is available. Other studies also investigated the transductive transfer learning, which assumes the target data is available but not labeled [20].

TABLE 1
SPOTTING THE RESEARCH GAP FOR FACIAL AFFECT.

Granularity \ Adaptation	Without user specificity	With user specificity
Frame-level	Generic [1][19][7]	Transfer from combined source [3] & multiple single-sources [2][4][20]
Segment-level	MIL [6][7][12][13]	Transfer from multiple subsets of sources (ours)

Despite the success of transfer learning, there has not been much effort into applying it for facial affect recognition, nor has there been much attention on addressing inter-personal differences. Chu et al. [3] showed the effectiveness of the transductive transfer learning approach, which re-weighs the source training samples most relevant to the target user. However, instance re-weighting and model adaptation require a computationally expensive optimization. Sangineto et al. [2] presented an alternative method that mitigates the computational cost of learning a personalized model, by using a regression function to identify the target model parameters based on the mapping from source subjects' data distribution into their classifier parameters. Zen et al. [4] further simplified the mapping by using support vectors for parameter transfer. Additionally, it has also been found that compared to transfer learning from a single combined source [22] (i.e. a combined set of data of all source subjects), learning from multiple single-sources [2][4] has a higher chance of identifying similar users, which should achieve a better transfer. It has also been shown that approaches that use the target data without considering the annotation may fail to achieve a correct adaptation. Chen et al. [20] compared transferring knowledge with and without using target data annotation. They demonstrated that inductive transfer learning with target data annotation outperformed its counterpart.

Personalization of facial affect at the segment level can be even more challenging, due to the uncertain and subjective connection between the overall affect label and a video segment (usually thousands of frames), and the inadequacy of annotated target data (e.g. only dozens of annotations per subject). Therefore, transferring from a set of individual models [2][4][20] may not be a good choice in real-use situations. Since the amount of one subject's training data is usually insufficiently small, each of these weak individual classifiers may suffer from overfitting.

Table 1 further categorizes the related facial affect studies based on the granularity of their training set annotation (frame- vs segment- level) and user adaptation methodology. A large body of prior work has been done on the generic classifier [1].

Recent research suggests knowledge adaptation is effective to accommodate individual differences [2][3][4][20] and multiple-instance learning useful to reduce annotation effort [6][7][12][13].

This paper proposes a framework to jointly solve these two issues. We use a variant of bootstrapping to transfer knowledge from groups of source subjects. While attempting to accentuate the weights of source data that is similar to the target user, we wish to maintain sufficient diverse data for each weak classifier. In contrast to learning from a set of individual classifiers, we coordinate a set of weak generic classifiers, each of which is learnt from data of a subset of the source subjects.

III. System Overview of Fast-PADMA

The objective of our approach, fast Personal Affect Detection with Minimal Annotation (fast-PADMA), is to rapidly learn a user-adaptive facial affect recognizer for practical use. Although the MIL techniques significantly reduce the annotation effort by requiring only one overall label per video segment, collecting sufficient individual data (multiple well-annotated segments) is still time-consuming [14]. We therefore attempt to accelerate personal model learning by exploring source knowledge from other training subjects. In addition, to ensure the application feasibility in real use, we avoid the computationally expensive optimization that is prevalently used in knowledge transfer.

The proposed affect recognizer is trained on video segments of different subjects. Only one segment-level annotation of self-reported affect is required per segment. Fig. 1 shows the process. Geometric facial features are automatically extracted frame-by-frame from the video segments (Row 1). AMIL is then used to calculate the overall association between facial gestures and affects in a segment for each individual user [14]. The AMIL result is then combined with descriptive statistics, such as the standard deviation and kurtosis of the temporal sequence of each facial feature, as the final feature representation for a segment (Row 2). The final feature vector therefore encapsulates (1) the indicativeness of the important facial gestures in a segment and (2) the overall attributes of the facial actions in the entire segment.

To alleviate the individual geometric bias, we introduce a personal data alignment technique that takes into account the "expressionless" states of the individual. We construct multiple *weak generic classifiers*, each of which is trained on *a subset of* the source subjects' data. Fast-PADMA then adapts to a particular target user by using the available target data to evaluate and re-weight each of the weak generic classifiers. As illustrated in Fig. 1 (Row 3), the weighting (indicated by varying intensities of blue) of each weak generic classifier can be different, according to the actual data of the target user. The final classification result depends on the weighted results of all of the weak generic classifiers.

IV. Modeling the Adaptive Facial Affect Model

Fast-PADMA is designed for scenarios where limited data from the target user is available. The assumption is that we have

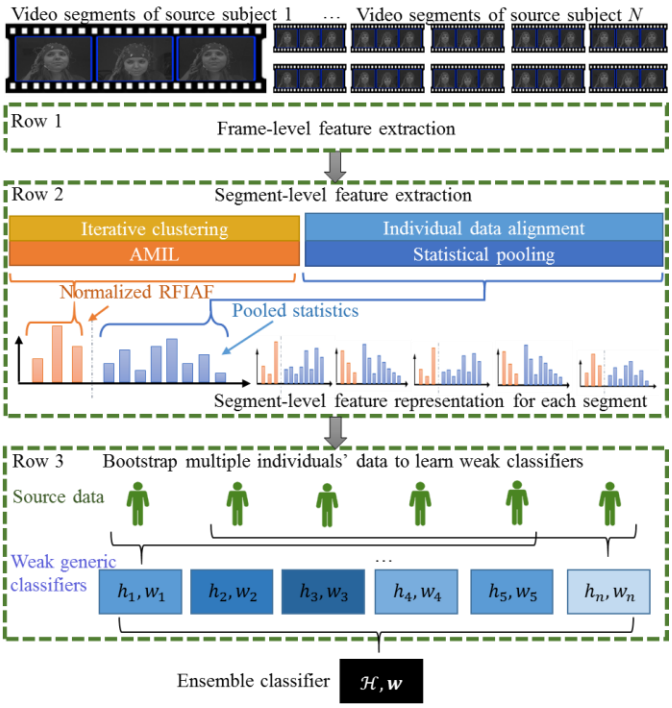


Fig. 1. System overview for fast-PADMA. Geometric facial features are automatically extracted from video segments. Individual data alignment is used to mitigate the personal geometric bias. Segment-level feature representation is obtained by AMIL and pooling. The ensemble classifiers are then trained on resampled data consisting of different combinations of *N*-1 subjects.

a number of video segments of N source subjects' facial expressions, and a smaller number of segments of the target subject. Each video segment j has a self-reported affect label y. The number of video segments available for subject u is denoted as $n^{(u)}$, the number of instances (frames) in video segment j from subject u is denoted as $n_j^{(u)}$, and the self-reported affect label for the segment is y_j^u.

From each frame in video segment j, a 20-dimension frame-level facial feature vector $\boldsymbol{x}$ is extracted. This gives us $\boldsymbol{X}_j^u = \{\boldsymbol{x}_i^u\}_{i=1}^{n_j^{(u)}}$ as the resulting set of feature vectors, and $D_u = \{\boldsymbol{X}_j^u, y_j^u\}_{j=1}^{n^{(u)}}$ as the data available from subject u. Taking this over N source subjects then gives us the training set $D^s = \{D_u^s\}_{u=1}^N$.

Likewise, for the target user t, we also have the target data $D^t = \{\boldsymbol{X}_j^t, y_j^t\}_{j=1}^{n^t}$, where n^t denotes the number of segments in D^t.

Our objective is to build an adaptive classifier to identify the label for a set of unseen instance frames of the target user $\boldsymbol{X}^t$ based on $D = D^s \cup D^t$, i.e. $f_T: \boldsymbol{X}^t \rightarrow y^t$.

A. Extracting Frame-level Geometric Facial Features

The first step in fast-PADMA is to extract the facial features from the video segments. Fast-PADMA follows previous work to use the Supervised Descent Method [24] and a 3D landmark model to extract 20 geometric facial features from each video frame (Table 2). This reduces each of the video segments to a

TABLE 2
GEOMETRIC FACIAL FEATURES USED IN OUR METHOD.

Index of geometric features	Implication	Measurement
1-4	Inner and outer brow movement	Distance between eye brow corner and corresponding eye corners (left & right)
5-6	Eye brow movement	Distance between the eye center and the corresponding brow center
7-8	Eye lid movement	Sum distance between corresponding landmarks on the upper and lower lid
9	Upper lip movement	Distance between the nose tip landmark and upper lip center landmark
10-11	Lip corner puller	Distance between the mouth corner and the corresponding eye outer center landmark
12	Eye brow gatherer	Distance between inner eye brow corners
13	Lower lip depressor	Distance between the chin bottom landmark and lower lip center landmark
14	Lip pucker	Perimeter of the mouth outer contour
15	Lip stretcher	Distance between the mouth corners
16	Lip thickness variation	Sum distance between corresponding points on the outer and inner contours
17	Lip tightener	Sum distance of corresponding points on the upper and lower mouth outer contour
18	Lip parted	Sum distance of corresponding points on the upper and lower mouth inner contour
19	Lip depressor	Angle between mouth corners and lip upper center
20	Cheek raiser	Angle between nose wing and nose center

sequence of 20-dimension facial feature vectors, $\{\boldsymbol{x}_i\}_{i=1}^{n}$, where *n* is the number of frames in the segment.

B. Transforming Data from Individuals into a Normalized Feature Space

Fast-PADMA is designed for scenarios in which there is a small amount of data from the target user. Our challenge is to utilize this labeled target data together with the source subjects' data to identify the personal attributes, so as to transform and align each individual's data to a normalized feature space.

In contrast to posed or simulated expressions, or expressions of pain, many spontaneous expressions of emotions are subtle in nature. Visually, the difference that results from facial expression changes is usually marginal, compared with the difference resulting from personal appearances. This means that when it comes to learning spontaneous facial affect from different source subjects' data, it is critical to perform individual data alignment to emphasize the emotion-induced facial deformation, and reduce identity bias.

We make the assumption that a similar physical movement, as reflected in the direction and magnitude of deformation of the geometric facial features, is caused by a similar implication of affect across subjects. In other words, we assume, for example, the maximum observed degree of mouth openness of different subjects indicates the same affect to a similar degree, such as equal intensity of surprise.

To achieve this, we need to align the feature boundaries (maximum and minimum) of different subjects to the same points in the normalized feature space. In addition, we also need to identify the default expressionless or *neutral* face from which all deformations are measured.

1) Identifying a neutral facial expression

Previous studies have shown that normalizing with respect to the expressionless or neutral frame is vital in removing subject-dependent bias [6][27]. A conventional method is to calculate the landmark displacement relative to the neutral frame. For instance, displacement features are obtained by subtracting the x and y coordinates of the facial landmarks in each frame from the corresponding coordinates in the first frame (neutral frame) in CK+ [27]. The rationale for this normalization is to construct a displacement feature space, where features share similar indicativeness to affect.

This feature normalization has two main drawbacks. First, without accounting for the individual feature boundaries, it assumes the displacement implication is identical across subjects. This assumption, however, is not valid due to the diversity of individualities. Second, it requires identifying the neutral frame. In datasets such as CK+, this is the first frame of each sequence. However, this is not always the case in real-use situations. A naïve assumption is that the neutral expression is the most frequent centroid in the centroid ID sequences. However, this is not true in most video-elicited datasets. For instance, a good number of amusing elicitation videos may lead to a majority of smile-containing frames in the dataset. Others may show obvious, emotional expressions towards the stimuli while the neutral expression does not show frequently. Furthermore, depending on the success of the emotion elicitation, the lengths of the stimuli also influence the expression distribution.

We propose a novel approach that makes use of the association between facial expression frames and the affect labels to identify the neutral frame from the weakly annotated data. We assume that neutral is *the expression that occurs most frequently across video clip-sets with different self-reported affect labels*. It is not difficult to see that this accords with the indicativeness of a facial gestures, as defined through the RFIAF measure [14], which takes into account both the prevalence of a gesture and its rarity over all affects.

Given the centroid ID sequences, we introduce the RFAF that measures the *non-indicativeness* of an expression. The RFAF of an expression c_j is defined as:

$$\mathrm{RFAF}(c_j, V) = \sum_{v_i \in V} \mathrm{RF}(c_j, v_i) * \mathrm{AF}(c_j, V) \tag{10}$$

RF is as previously defined, and AF measures the generality of c_j among different affects:

$$\mathrm{AF}(c_j, V) = log \frac{\left|\{v \in V : f(c_j, v) > 0\}\right|}{1 + |V|} \tag{11}$$

By selecting the facial centroid with the maximum RFAF value to represent the neutral frame $\boldsymbol{x}^{(n)}$:

$$\boldsymbol{x}^{(n)} = \mathrm{argmax}_{c_j} \mathrm{RFAF}(c_j, V) \tag{11}$$

we then have $\boldsymbol{x}^{(n)} = \langle x_1^{(n)}, \ldots, x_{20}^{(n)} \rangle$, where each $x_i^{(n)}$ denotes

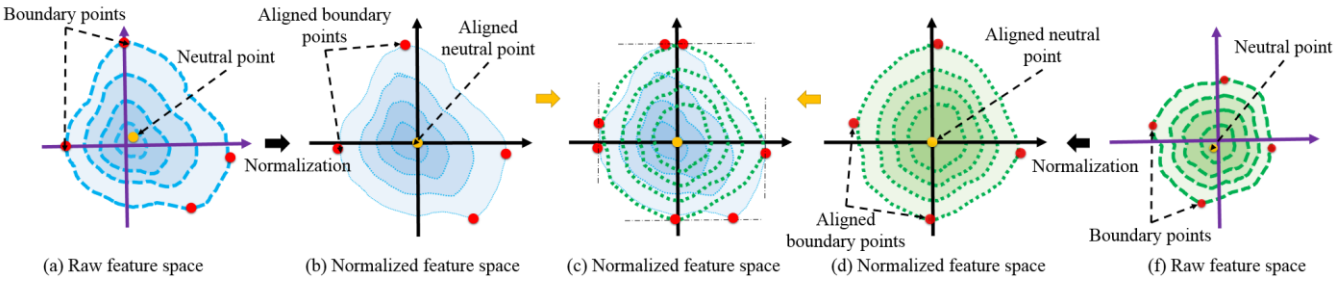


Fig. 2. Illustration of data alignment from data of two subjects. The purple axes in (a) and (f) indicate the 2D projection of each subject's data in the raw feature space before alignment. Each inner contour represents a different level of data density. Red and yellow dots denote the boundary values and the neutral point, respectively. (b) and (d) show the transformed distributions to the normalized feature space of (a) and (f), respectively. Putting (b) and (d) in the same axes (c) aligns the boundaries of the two subjects with the same values and their neutral points to the same point in the center of the normalized feature space.

the value of a geometric facial feature in the neutral frame of a particular subject.

2) Aligning data according to neutral and boundary values

The next step is to align the data with respect to the neutral expression, and the boundary values. For each feature, we first use unity-based normalization [26] to align feature boundaries (minimum and maximum) of the data from all subjects to the same points in the normalized feature space. The result is that the data from all individuals are comparable and have the same scale.

The next step normalizes with respect to the neutral frame. We define a mapping function to align the geometric features,

$$\hat{\boldsymbol{x}} = \phi_a(\boldsymbol{x}) \tag{12}$$

The function ϕ_a normalizes each feature from the individual data by a piecewise function, leaving the complex non-linear transformation to be learnt by the supervised classifier. For the i-th geometric facial feature x_i, we align it to

$$x_i' = q_0(x_i)^{\theta} q_1(x_i)^{1-\theta} \tag{13}$$

where θ is a Heaviside step function:

$$\theta = \frac{1}{2}(1 + \text{sign}(x_i - x_i^{(n)})) \tag{14}$$

$q_0(\cdot)$ and $q_1(\cdot)$ are the scaling functions:

$$q_0(x_i) = \frac{1}{2} \cdot \frac{x_i - x_i^{(n)}}{x_{i,max} - x_i^{(n)}} + \frac{1}{2}$$
$$q_1(x_i) = \frac{1}{2} \cdot \frac{x_i - x_{i,min}}{x_i^{(n)} - x_{i,min}} \tag{15}$$

where $x_{i,min}$ and $x_{i,max}$ represent the minimum and maximum values of i-th geometric facial feature across all segments of a particular subject.

Post individual data alignment, (8) and (9) can be rewritten as:

$$\hat{\boldsymbol{z}}_s = \phi_s(\phi_a(\boldsymbol{x}_j) | \boldsymbol{x}_j \in \boldsymbol{X}) \tag{16}$$

and

$$\boldsymbol{z} = \langle \hat{\boldsymbol{z}}_a, \hat{\boldsymbol{z}}_{s_i} | s_i \in S \rangle \tag{17}$$

Fig. 2 illustrates the result of the alignment process. The purple axes in (a) and (f) indicate the 2D projection of two subjects' data in the raw feature space before alignment. Each inner contour delineates a different level of data density. Red and yellow dots denote the boundary values and the neutral point, respectively. It can be seen from the figures that the neutral point is not necessarily the densest point in the distribution. Aligning each individual's data independently according to (12) transforms the distributions to the normalized feature space as shown in Fig. 2 (b) and (d). The general shapes of the transformed data distribution remain highly consistent with the raw distributions, however, the boundaries are aligned to the same values across the two subjects. The two distributions fall into the same bounding box in Fig. 2 (c), and the neutral points of different subjects are aligned to the center of the normalized space.

Unity-based normalization has been used by Soleymani et al. [26] to reduce the differences among participants. However, their method uses just two points for alignment (min & max), in essence assuming that the deformation is linear throughout. Our method utilizes one more fiducial point, i.e. the neutral point, to align the distributions of different individuals. In this sense, we explicitly account for the geometry of the neutral facial expression for our classifiers.

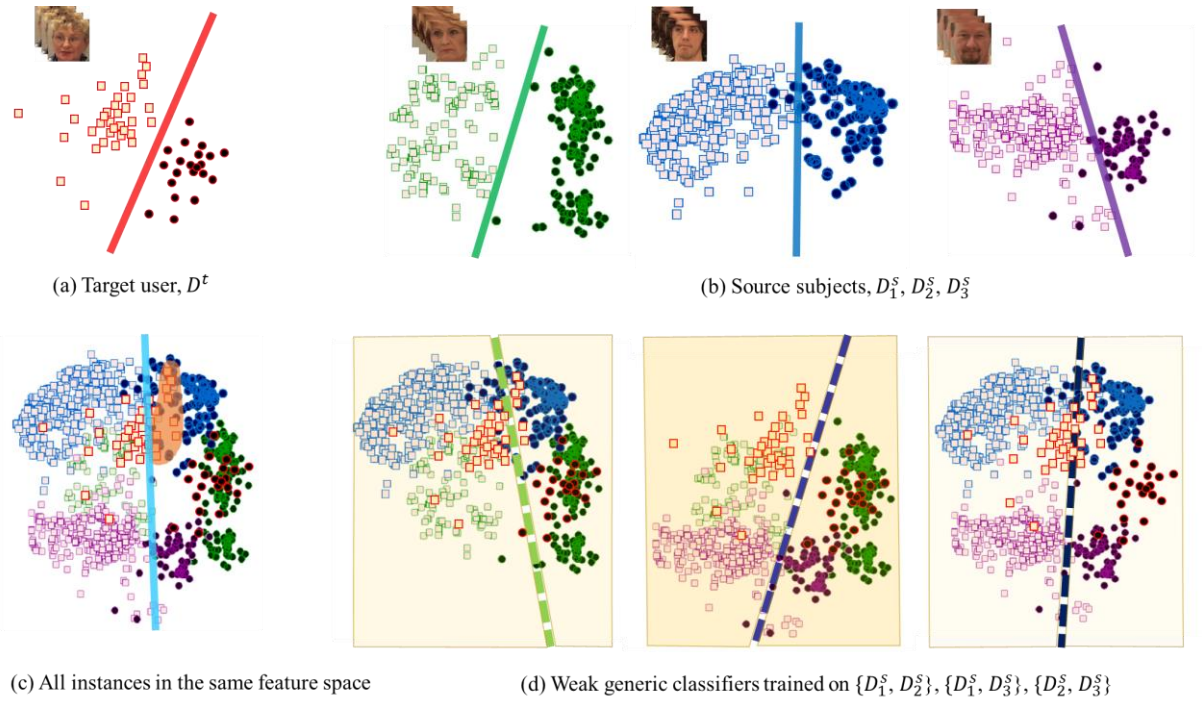
C. Building the Segment-Level Data Instances

Given the sequence of raw and normalized facial feature vectors, the next step in fast-PADMA is to construct the data instances that will be used in the training and testing process.

Each data instance in fast-PADMA corresponds to one video segment. Our instances are designed to capture (1) the level of association of the segment with all possible affects, and (2) the statistical information describing the facial features from the video frames.

Fast-PADMA follows a process from previous work [14], which first uses an iterative k-means clustering process to identify similar *facial expressions* from the frame-level feature vectors. The feature vectors are then replaced with their cluster labels, which represent the expression that is currently exhibited by the user. The sequence of cluster labels is then mined for *facial gestures*, using a frequent subsequence mining process in a temporal moving window [5][6][14].

Once the frequently-occurring facial gestures have been extracted, Association-based Multiple Instance Learning (AMIL) is used to calculate the overall affect association of the extracted gestures. In contrast to the conventional MIL methods that select some instance(s) as prototype(s) to represent a bag, AMIL explores all potential indicative instances. It assumes that if an instance (e.g. a facial gestures) occurs frequently in segment(s) labeled with one particular class (e.g. a reported user



(a) Target user, D^t

(b) Source subjects, D_1^s, D_2^s, D_3^s

(c) All instances in the same feature space

(d) Weak generic classifiers trained on $\{D_1^s, D_2^s\}$, $\{D_1^s, D_3^s\}$, $\{D_2^s, D_3^s\}$

Fig. 3 Illustration of the data projections in the proposed method. (a) and (b) present the data distribution of the target user and source subjects, respectively. Positive and negative instances are indicated by the squares and circles. The solid lines in (a) and (b) denote the ideal hyperplanes of each user-specific model. (c) shows the projection of the source and target data in the same feature space. The orange shadow highlights the conflicting instances between D^t and D_2^s, which will be misclassified by the generic classifier learnt directly from all the training instances. (d) demonstrates the weak classifiers learnt on the bootstrapped data, each of which excludes one different source subject. The dash lines indicate the hyperplanes of the weak generic classifiers. The background transparency denotes the corresponding weight of the weak generic classifier, which depends on the performance on the available target set.

affect), but not in others, this instance has a strong association with that class.

Following the AMIL procedure [14], we calculate the RFIAF value to reflect the affect implication of a gesture g_j:

$$\mathrm{RFIAF}(g_j, v_i, V) = \mathrm{RF}(g_j, v_i) * \mathrm{IAF}(g_j, V) \quad (3)$$

The first half of the formula is the *response frequency*

$$\mathrm{RF}(g_j, v_i) = \frac{f(g_j, v_i)}{max\{f(g, v_i): g \in G\}} \quad (4)$$

which measures the prevalence of gesture g_j over the clip-set v_i. v_i is defined as the set of video segments that have been self-reported by the user to exhibit the affect annotation a_i, and $f(g_j, v_i)$ denotes the occurrence frequency of gesture g_j in v_i.

The second half of the formula represents the *inverse affect frequency*

$$\mathrm{IAF}(g_j, V) = log \frac{1 + |V|}{|\{v \in V: f(g_j, v) > 0\}|} \quad (5)$$

which quantifies the indicativeness of g_j by measuring its "rarity". V is the set that contains all video clip-sets from the subject; $|V|$ denotes the number of different self-reported affects, and $|\{v \in V: f(g_j, v) > 0\}|$ represents the number of clip-sets that contain g_j.

Given the RFIAF values between each gesture and affect, we can construct an instance vector $\boldsymbol{z}_a$ to represent the segment-affect association of a video segment by aggregating over all the gestures that are contained in a segment. Each element z_{ai} in $\boldsymbol{z}_a$ indicates the association of the segment with a particular affect,

$$z_{ai} = \sum_{t=s/2}^{n-s/2} \sum_{g_j \in G_t} \mathrm{RFIAF}(g_j, v_i, V) \quad (6)$$

G_t indicates the set of gestures occurring in the moving window spanning over s+1 frames with its center at the t-th frame of a segment; n is the number of frames in a segment; v_i is the clip-set corresponding to affect a_i; and V denotes all available clip-sets from the subject . We then calculate the normalized $\boldsymbol{z}_a$, which gives us the association between the segment and all possible affects:

$$\hat{\boldsymbol{z}}_a = \frac{\boldsymbol{z}_a}{|\boldsymbol{z}_a|} \quad (7)$$

D.Pooling Frame-level Statistics at the Segment Level

Although $\hat{\boldsymbol{z}}_a$ can capture the *overall* affect associations of the segment, it does not reflect characteristics such as instance dispersion and distribution. We believe that these characteristics are informative as they capture the *structure* of the segment.

Similar to previous work [6], we use pooling to aggregate frame-level features for segment-level representation. Denoting ϕ_s as the pooling operation of different statistical descriptors, we define

$$\boldsymbol{z}_s = \phi_s(\boldsymbol{x}_j | \boldsymbol{x}_j \in \boldsymbol{X}) \quad (8)$$

as the resulting feature vector of the statistical pooling operation. The suggested descriptors ϕ_s can include dispersion attributes such as mean, median, standard deviation, variance, minimum and maximum, and the distribution attributes such as skewness and kurtosis.

Since the frame-level feature vectors are in a 20-dimension

space $\boldsymbol{\mathcal{X}} \in \Re^{20}$, concatenating the segment statistics features $\boldsymbol{z}_s$ will increment the segment vector $\boldsymbol{z}$ by 20 additional dimensions for each descriptor ϕ_s. This suggests that we need to avoid using redundant descriptors as this will create a high dimensionality problem. In pilot experiments, we observe that the pooled features of mean and median have similar contribution with respect to the final recognition, as do those of standard deviation and variance. We therefore exclude the descriptors of median and variance to reduce vector dimensionality in the final model.

Our final segment feature representation is therefore

$$\boldsymbol{z} = \langle \hat{\boldsymbol{z}}_a, \boldsymbol{z}_{s_i} | s_i \in S \rangle \quad (9)$$

where s_i indicates a particular statistical descriptor in the set of descriptors S, which include mean, standard deviation, minimum, maximum, skewness and kurtosis. In other words, it is a concatenation of the normalized segment-affect association vector $\hat{\boldsymbol{z}}_a$ and the pooled statistics vectors.

E. Building an Ensemble Classifier

The final step in fast-PADMA is to build an ensemble classifier for spontaneous facial affect recognition. Unlike the conventional generic classifier that is trained on all the source subjects' data, D^s, the fast-PADMA process learns a set of classifiers, each of which is trained on a subset of D^s. More specifically, given *N* source subjects, we train *N* weak generic classifiers using a different subset of data from $N-1$ subjects for each classifier. Compared with the classical approach of building ensemble classifiers which combines classifiers, each of which are trained on data from one individual source subject, each classifier in our approach is trained from more data and less likely to suffer from overfitting.

Our hypothesis is that there are some conflicting instances between the target and source subjects that are located closely in the feature space, but with contradictory annotation. Fig. 3 (a) and (b) show an illustration of the 2D data projection of the target user and three source subjects. Although positive and negative instances of the target user appear to be nicely separable in (a), projecting all these data to one space (c) results in some of the target instances (in the orange region) having conflicting annotations with instances in D_2^s, which makes it learning a hyperplane problematic. Fig. 3 (d) shows three classifiers trained on different data combinations of the source subject. It is clear that removing the source subject that contains the conflicting instances will lead to a clear discrimination of the target data.

The difficulty with this approach is that the conflicting instances are generally spread out across different source subjects and it is impractical to identify all these instances in advance. Additionally, the presence of conflicting instances does not negate the contribution of other instances from the same source subject. Therefore, rather than completely discard a data subset, we retain all the weak generic classifiers and weigh them based on their performances on the available target data. This is illustrated in Fig. 3 (d) where the background intensity corresponds to the weight of the weak generic classifiers on the available target set. It is clear that the more separable datasets, which would presumably be easier to learn and therefore achieve better performances, are is given higher weights by the final ensemble classifier.

Given a source set D^s and a target set D^t, we propose to learn an ensemble classifier for the target user and infer the label y^t for an unseen target set $\boldsymbol{X}^t$. The algorithm is summarized in Table 3. We extract $\boldsymbol{z}^t$ as the segment representation of $\boldsymbol{X}^t$ and predict the label y^t according to a weighted voting mechanism:

$$\text{argmax}_{y^t} \sum_{n=1}^{N} w_n I(h_n(\boldsymbol{z}^t) = y^t) \quad (18)$$

where $I(\cdot)$ is an indicator function and w_n is the weight for a weak generic classifier. In our experiments, we use a linear sequential minimal optimization (SMO) [28] to build the weak classifiers, whose cost parameter is set to 0.05.

In addition to reducing the impacts from the conflicting instances and overfitting, fast-PADMA has advantage over the transfer technique for its simplicity and efficiency. To apply our method, we only need to evaluate the performance of the weak generic classifiers on the available target set, without any computational distribution fitting or similarity calculation. For the same reason, the source dataset, which is often huge or which contains personal information, need not be deployed, stored or used for retraining in the real-use situation. This is essential for the rapid building and feasibility of the target facial affect model.

TABLE 3
LEARNING THE ENSEMBLE CLASSIFIER.

Input: source set D^s and the training set of the target subject D^t
Output: A set of weak generic classifiers $\mathcal{H}$ for the target user and the corresponding weights $\boldsymbol{w}$
Phase-I Learning weak classifier set $\mathcal{H} = \{h_i\}$ **from** $D^s = D_1^s, \ldots, D_N^s$ **for** n = 1 to N Resample a subset of training data from source set, $D_n = D^s \backslash D_n^s$ Learn a weak generic classifier h_n on D_n Update the weak classifier set $\mathcal{H} \leftarrow \mathcal{H} \cup h_n$ **end for**
Phase-II Learning an ensemble classifier on target training set D^t **for** n = 1 to N Evaluate h_n on D^t, classification error $\varepsilon = \sum_{j=1}^{n^t} y_j^t \neq h_n(\boldsymbol{z}_j^t)/n^t$ Update weights $w_n = (1-\varepsilon)/\varepsilon$, $\boldsymbol{w} \leftarrow \boldsymbol{w} \cup w_n$ **end for**
return $\{\mathcal{H}, \boldsymbol{w}\}$

V. EXPERIMENTAL EVALUATION

Given a new user, fast-PADMA (1) aligns the target user data with data from our source subjects using the neutral point and the boundary values; (2) extracts the segment-level feature representation; and (3) constructs an ensemble classifier from data of source subjects who are similar to the target user. Our evaluation therefore presents substantiated experimental results for these 3 aspects and shows the effectiveness of fast-PADMA by comparing against user-specific, generic and hybrid models.

For fairness in evaluation, all models use the same machine learning classifier (SMO) and the features are kept constant. The only exception is the comparison against the generic model. Since AMIL relies on the availability of a bag label to obtain the associations between expressions and affects, but the

generic model does not have access to target user data, segment features extracted by AMIL from the target data were excluded from the ensemble model in this comparison.

The protocol for evaluation is as follows: for models trained using target user data, performance is measured via leave-one-segment-out cross-validation; for the generic classifier trained without target data, performance is evaluated via leave-one-subject-out cross-validation. The reported result is the weighted average across segments and subjects.

A. Evaluation Data

We evaluate fast-PADMA on the binary pain/non-pain classification on the UNBC-McMaster Shoulder Pain Expression Archive Database (UNBC) [19], and 3-class arousal and valence classifications on three datasets: Denver Intensity of Spontaneous Facial Actions (DISFA) [29], Mobile Spontaneous Affect Response Video (MSARV) [14] and MAHNOB-HCI emotion recognition dataset (MAHNOB) [26]. These datasets cover a diverse range scenario.

1) UNBC

UNBC contains 200 segments of near frontal facial expressions from 25 subjects with potential shoulder pain. The length of each segment ranges from 58 to 518 frames. Expert coders used Observer Pain Intensity (OPI) rating to score the segments in 6-point scale (0: no pain; 5: strong pain). Following the protocol of previous studies [6][7][12], we re-define the binary segment-level label with OPI≥3 as "pain" and OPI=0 as "no pain" and omit the segments with intermediate intensity. Excluding subjects with only one resulting segment gives us 23 subjects and 147 segments.

2) DISFA

The DISFA dataset consists of 27 subjects. Each subject has 9 frontal spontaneous response segments to a set of emotional stimuli. The total length of the stimuli is around 4 minutes. The original annotation of DISFA is the frame-level activation level of the facial AUs. To acquire the segment-level affect annotation, we recruited 6 postgraduate students aged 21-31 (3 female) to manually identify the affects appearing in each segment, including anger, disgust, fear, happiness, sadness, surprise and neutral if none of above appears. The annotated results across different observers show a high degree of agreement with an average Fleiss' Kappa of 0.606 across all emotion categories.

3) MSARV

MSARV is composed of 11 subjects' frontal facial response to emotion stimuli, which were collected with the front camera of a mobile phone. Compared to DISFA, each subject has a large amount of data (25 segments). The total length of the stimuli is around 41 minutes. The original segment annotation includes neutral, happiness, interest, boredom, sadness, disgust, fear and anger. As this work focuses on arousal and valence classification and there is no clear mapping from boredom and interest into arousal and valence, we therefore exclude segments that are labelled as one of these two affects. This gives us 11 subjects and 106 remaining segments.

4) MAHNOB

The experimentation of MAHNOB contains two distinct paradigms, the emotion experiment (affect classification) and the implicit tagging (agreement/disagreement classification). We evaluate on the former, as it is consistent with the other datasets in the sense of using video stimuli to elicit spontaneous responses. Their data collection uses multiple cameras for recording. We used the "close up from the bottom right" view since some subjects' faces were frequently obscured in the frontal camera view. The self-reported emotion category includes neutral, anxiety, amusement, sadness, joy, disgust, anger, surprise and fear. Removing subjects with incomplete annotation gives us 27 subjects. Each subject has 20 segments, which in total last around 40 minutes.

TABLE 4
MAPPING EMOTION INTO THREE CLASSES ON AROUSAL AND VALENCE [26]

Arousal classes	Emotion
Calm/Low arousal	Sadness, disgust, neutral
Medium arousal	Joy and happiness, amusement
Excited/High arousal	Surprise, fear, anger, anxiety

Valence classes	Emotion
Unpleasant/Low valence	Fear, anger, disgust, sadness, anxiety
Neutral/Medium valence	Surprise, neutral
Pleasant/High valence	Joy and happiness, amusement

B. Mapping Emotions into Arousal and Valence

The annotated emotional categories vary across the datasets. For the purpose of analysis and comparison, we follow previous work [26][30] to convert the categorical emotions into arousal and valence (see Table 4). This defines two 3-class classification problems for DISFA, MSARV and MAHNOB, i.e. the classifications of low, medium, and high levels of arousal and valence, respectively.

C. Experiment Results

We first demonstrate the effectiveness of our feature representation by benchmarking fast-PADMA against the state-of-the-art performances on UNBC and MAHNOB. We then compare different alignment techniques and specifically evaluate the ensemble mechanism across all 4 datasets. This is followed by a summary of different learning paradigms, which shows the advantage of the proposed user-adaptive model.

1) External comparison on UNBC and MAHNOB

Since most of the pertinent MIL techniques proposed for facial affect recognition were evaluated on UNBC, we first compare fast-PADMA against the state-of-the-art in the user-independent learning paradigm. The AMIL related features are excluded in this evaluation paradigm, due to unavailability of the target data. The performance is measured by the accuracy at the equal error rate, following previous practice [12][31].

Table 5 shows the results. Fast-PADMA, which learns from the pooled statistics features of the aligned data, achieves performance equivalent to the best reported state-of-the-art performance (85.7) on UNBC. This result implies that (1) in spite of their simplicity, well-designed geometric facial features can give a high classification performance for pain detection, which suggests that geometric features have yet not been fully

TABLE 5
PERFORMANCE AND COMPARISON TO STATE-OF-THE-ART MIL METHODS ON USER-INDEPENDENT LEARNING, UNBC DATASET (PERFORMANCE METRIC: ACCURACY AT EQUAL ERROR RATE)

MIL-Boost[5]	MILIS [9]	MILES [8]	MS-MIL [31]	AMIL [14]	RMC-MIL [12]	fast-PADMA (ours)*
76.9	76.9	78.2	83.7	84.4	**85.7**	**85.7**

*Segment feature representation in the experiment contains the pooled statistics features only. AMIL is not used due to the unavailability of target data.

TABLE 6
PERFORMANCE AND COMPARISON TO STATE-OF-THE-ART MIL METHODS ON USER-INDEPENDENT LEARNING, MAHNOB DATASET (PERFORMANCE METRIC: ACCURACY AT EQUAL ERROR RATE)

Arousal			Valence		
EPFKNN [13]	EPFSVM [13]	fast-PADMA (ours)*	EPFKNN [13]	EPFSVM [13]	fast-PADMA (ours)*
46.4	53.6	**58.4**	44.1	50.6	**53.9**

*Segment feature representation in the experiment contains the pooled statistics features only. AMIL is not used due to the unavailability of the target data.

explored [1]; and (2) the proposed aggregation technique ϕ_s, which extracts the segment attributes from frames by pooling descriptive statistics features, is simple, fast and effective.

Compared with the UNBC dataset, few studies have been conducted on MAHNOB, probably due to its challenges – the spontaneous affects it contains are more subtle and complex. We therefore compare our result with the latest published results [13], which follows established protocol [26] to map the categorical emotions into three classes for arousal and valence.

Table 6 shows our comparison results. As indicated by the bolded numbers, fast-PADMA outperforms the k-NN and SVM models [13] in both arousal and valence classifications on MAHNOB. This further validates our techniques of segment feature extraction and data alignment.

2) Evaluating alignment and adaptation of data from multiple individuals

To thoroughly evaluate fast-PADMA, this section provides in-depth comparisons across alignment techniques and learning paradigms. Fig. 4 presents the comparison across the 4 datasets. The performance of each alignment technique is denoted with a line of a different color. The vertical axis shows the correctly classified rate accuracy. The horizontal axis presents the composition of the training data. "Specific" indicates that the model is trained only on data from the target user. "+20%" to "+100%" indicates models trained on a mix of source subject and target data, with the percentage representing the percentage of source subject data in the entire training set. Selection of the source data is determined by random resampling. "Generic" means the generic classifier trained on source subjects' data alone.

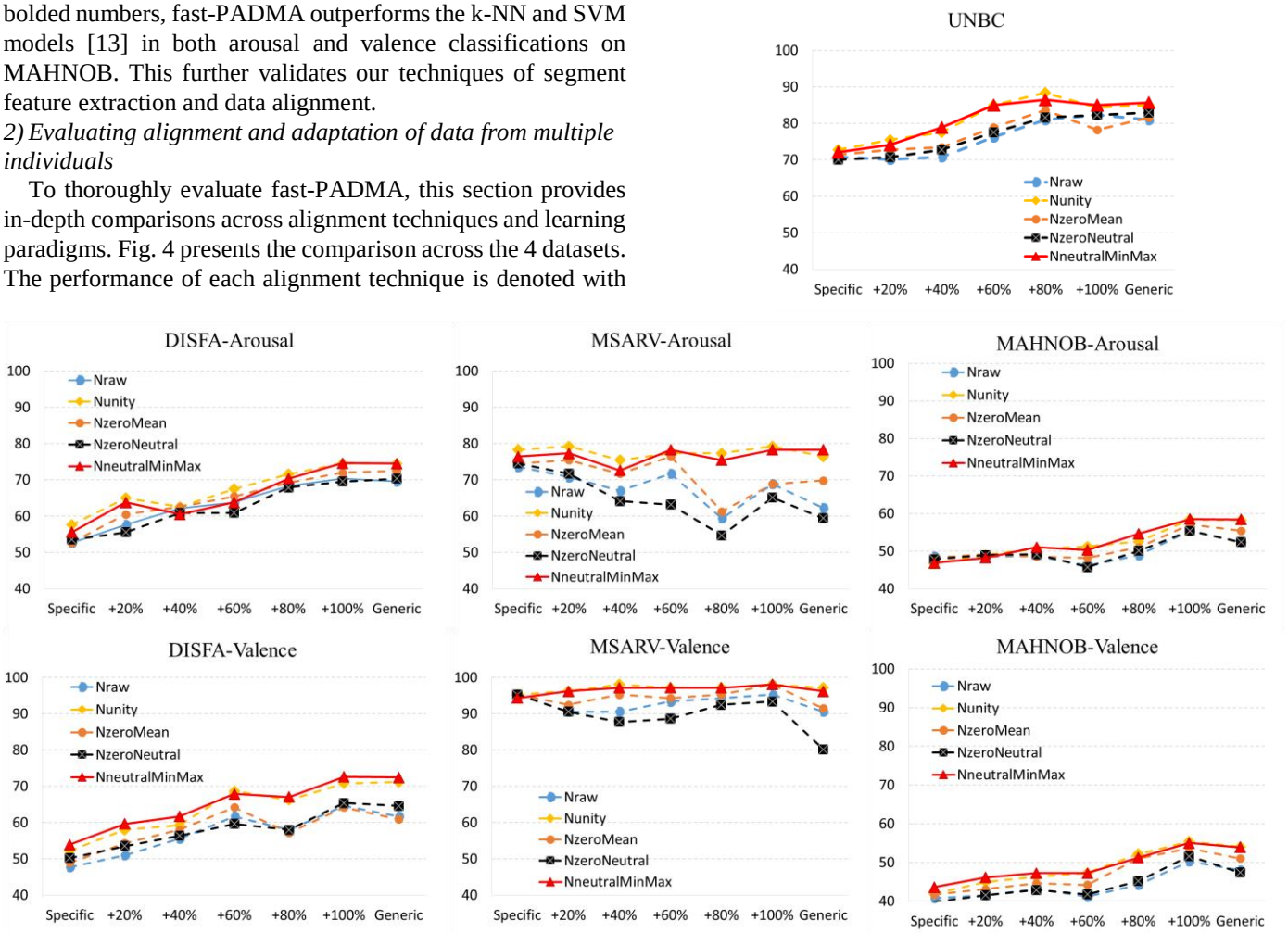


Fig. 4 Performance in correctly classified rate of pain/no pain classification on UNBC and three classes' arousal and valence classifications on DISFA, MSARV and MAHNOB datasets. Lines with different colors represent the techniques for data alignment. The vertical axis denotes the performance and the horizontal axis the models learnt on different sets of data. "Specific" indicates the model learnt on the target data only. "+20%", "+40%", "+60%", "+80%" and "+100%" denote the models trained on the available target data plus the particular percentage of the source data. "Generic" means the generic classifier built without the use of target data. The overall results show that the proposed data alignment technique (in red) generally performs well among the alignment techniques. In addition, although variations exist among datasets and classification problems, performance usually peaks at hybrid models learnt from target data plus a certain amount of source data.

Fig. 6. Examples of the identified neutral frames from different datasets.

For simplicity, we refer to the various modes of alignment as
(a) $\mathcal{N}_{\text{raw}}$: raw representation (no normalization);
(b) $\mathcal{N}_{\text{unity}}$: uses unity-based normalization to transform each feature of individual subject to [0,1];
(c) $\mathcal{N}_{\text{zeroMean}}$: shifts the values of each feature with its mean aligned to zero;
(d) $\mathcal{N}_{\text{zeroNeutral}}$: shifts each feature with its value at the neutral frame aligned to zero;
(e) $\mathcal{N}_{\text{neutralMinMax}}$: our previously-presented alignment method, which simultaneously accounts for the neutral frame and the boundaries.

Likewise, we use the following abbreviations for models:
(a) $\mathcal{M}^{\text{single}}_{\text{target+source}}$: a single classifier, trained on the available target data and all source subjects' data;
(b) $\mathcal{M}^{\text{single}}_{\text{target+\%source}}$: a single classifier, trained on the available target data and some of the source subjects' data. For simplicity and efficiency (tractable running time), we select the portion of the source subjects' data as the first $\langle 0, 20, 40, 60, 80, 100\rangle$% of the available dataset;
(c) $\mathcal{M}^{\text{ensemble}}_{\text{individual}}$: an ensemble classifier consisting of N classifiers, each trained on data from one source subject;
(d) $\mathcal{M}^{\text{ensemble}}_{\text{generic}}$: an ensemble classifier composed of N classifiers, each trained on data from $N-1$ source subjects.

An overall comparison between alignment techniques shows that the $\mathcal{N}_{\text{neutralMinMax}}$ normalization (in red) generally outperforms the others, achieving the highest accuracies on arousal for DISFA (74.6%) and MAHNOB (58.5%) and valence for DISFA (72.6%) and MSARV (98.1%). This is especially true when compared with $\mathcal{N}_{\text{raw}}$, $\mathcal{N}_{\text{zeroMean}}$, and $\mathcal{N}_{\text{zeroNeutral}}$, notably for the models which have access to an additional 80% of source data on DISFA-valence and MSARV-arousal. Fig. 6 shows some examples of identified neutral frames from different datasets.

Learning from the incremental data aligned by Nubn normally leads to a steady increase in accuracy. In other words, as we increase the training data from different subjects (observed from the first six data points), the performance of the data alignment with Nubn increases monotonically. This trend suggests that, given a good alignment, data from different source subjects can be used together in one generic classifier. In contrast, improper alignment of the data from different subjects may hurt the performance of a classifier due to inter-personal differences. Examples can be seen at the performance fluctuation at "+80%" in MSARV-arousal and the decrease from "+20" to "+40%" in MSARV-valence classification.

3) Evaluating the ensemble mechanism of fast-PADMA

The aforementioned results validate our segment feature representation and individual data alignment. The results also suggest that good performance can be achieved by using a hybrid model learnt from both target and source data. It also appears that the method of combining data -- or the amount of source data used for training, is key. This section investigates using an ensemble of weak generic classifiers as an alternative.

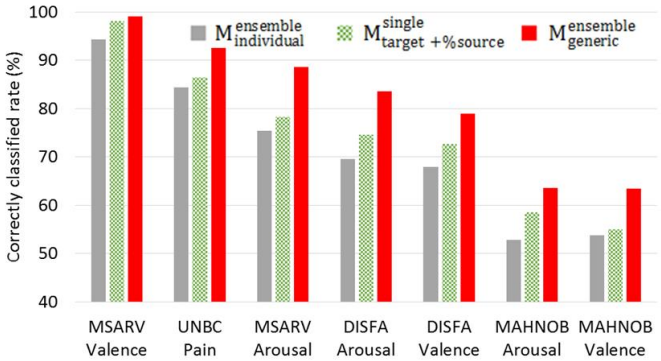


Fig. 5 Performance comparison in correctly classified rate: performance of the ensemble classifier of N weak individual classifiers ($\mathcal{M}^{\text{ensemble}}_{\text{individual}}$), best possible performance trained on all target data and some percentage of source data ($\mathcal{M}^{\text{single}}_{\text{target+\%source}}$), performance of the ensemble classifier of N weak generic classifiers ($\mathcal{M}^{\text{ensemble}}_{\text{generic}}$). Across different datasets, $\mathcal{M}^{\text{ensemble}}_{\text{generic}}$ in general outperforms other counterparts, which indicates the effectiveness of the proposed ensemble technique.

Fig. 5 compares performance across learning paradigms using the same data alignment technique ($\mathcal{N}_{\text{neutralMinMax}}$). Previous studies have explored knowledge transfer from single sources [2][4][20]. We use the ensemble of weak individual classifiers ($\mathcal{M}^{\text{ensemble}}_{\text{individual}}$) as a comparatively similar method.

It is encouraging that the method used by fast-PADMA, $\mathcal{M}^{\text{ensemble}}_{\text{generic}}$, surpasses $\mathcal{M}^{\text{ensemble}}_{\text{individual}}$ by 8.2% on UNBC, over 10% for arousal on DISFA (14.0%), MSARV (13.2%) and MAHNOB (10.8%), and 11.1%, 4.7%, 9.7% respectively for valence. Even better, $\mathcal{M}^{\text{ensemble}}_{\text{generic}}$ can constantly outperform $\mathcal{M}^{\text{single}}_{\text{target+\%source}}$, the best hybrid model trained on both the target and source data, by 6.57% on average.

For a more thorough analysis, we compare $\mathcal{M}^{\text{ensemble}}_{\text{generic}}$ with other ensemble learning methods from previous work. Common ensemble learning methods include bagging, boosting, random subspace, and stacking. Bagging, also known as bootstrap aggregation, trains multiple models on randomly drawn training subsets and aggregates them with equal weight. Boosting incrementally trains and reweighs the previous misclassified instances. Random subspace method combines classifiers learnt from the subspaces of the original feature space. Stacking mingles results from different types of classifiers. While Adaboost [23] is a classic boosting algorithm, random trees is a well-used random subspace method, and random forest [32] is the bagging version of random trees. We therefore compare $\mathcal{M}^{\text{ensemble}}_{\text{generic}}$ with the following ensemble methods: (1) bagging and (2) Adaboost with SMO, (3) random tree and (4) random forest, and (5) stacking of SMO & random forest, (6) SMO & k-NN [33], and (7) random forest & k-NN. The SMO classifiers in bagging, Adaboost, and stacking are

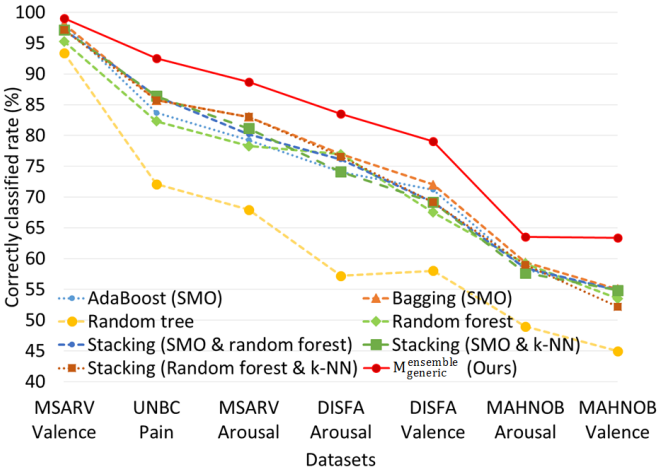


Fig. 7. Performance comparison of different ensemble learning classifiers. The vertical axis denotes the correctly classified rate. The horizontal axis shows the classification issues on different datasets, which are ranked according to the average performance.

configured with identical parameters as in the previous evaluation.

Fig. 7 shows the performance of $\mathcal{M}_{\text{generic}}^{\text{ensemble}}$ and the other ensemble learning methods on 4 datasets. Except for the random tree, which shows relatively poor performance, the performance of the different ensemble methods are similar. It is encouraging to observe that the performances achieved by $\mathcal{M}_{\text{generic}}^{\text{ensemble}}$ are in general considerably higher than other ensemble methods: on average, $\mathcal{M}_{\text{generic}}^{\text{ensemble}}$ outperforms the second-best classifiers by 5.5%. The only marginal match comes from the classification of MSARV-valence, where the baseline is originally very high.

4) Summarizing the comparison across learning paradigms

For a comprehensive understanding of the impact of different data compositions and training methods, Table 7 summarizes the performances of 4 highlighted models over the majority guess baseline. The comparison was conducted among the user-specific classifiers, generic classifiers, the best performing hybrid model ($\mathcal{M}_{\text{target+\%source}}^{\text{single}}$) and the ensemble classifier of multiple weak generic classifiers ($\mathcal{M}_{\text{generic}}^{\text{ensemble}}$). As expected, all learning paradigms outperform the baseline.

Interestingly, the generic classifiers generally reach considerably higher performances than the user-specific classifiers, except for a few close matches on MSARV (arousal: 78.3% vs 76.4% and valence: 96.2% vs 94.2%). This may be because the amount of available target data is modest and our proposed data alignment helps the generic classifier to accommodate identity bias. As expected, performances of $\mathcal{M}_{\text{target+\%source}}^{\text{single}}$ surpasses those of the user-specific and generic classifiers. This corroborates our idea that using both target and source data can facilitate affect modeling. The difficulty here would be to decide the amount of source data to incorporate into $\mathcal{M}_{\text{target+\%source}}^{\text{single}}$.

The most encouraging result is that of the ensemble classifier $\mathcal{M}_{\text{generic}}^{\text{ensemble}}$. $\mathcal{M}_{\text{generic}}^{\text{ensemble}}$ outperforms $\mathcal{M}_{\text{target+\%source}}^{\text{single}}$ by a remarkable 10.2% on average. This verifies our assumption that bootstrapping multiple weak generic classifiers can establish a high-performing adaptive model in an efficient manner.

TABLE 7
PERFORMANCE COMPARISON BETWEEN DIFFERENT LEARNING PARADIGMS: SPECIFIC, GENERIC AND HYBRID MODELS
(PERFORMANCE METRIC: CORRECTLY CLASSIFIED RATE)

Dataset / Model	UNBC	DISFA		MSARV		MAHNOB	
	Pain	A	V	A	V	A	V
Baseline	62.8	45.3	35.8	39.2	69.5	44	39.3
Specific	72.1	55.6	53.9	76.4	94.3	46.9	43.6
Generic	85.7	74.5	72.4	78.3	96.2	58.4	53.9
$\mathcal{M}_{\text{target+\%sour}}^{\text{single}}$	86.5	74.6	72.6	78.3	98.1	58.5	55.0
$\mathcal{M}_{\text{generic}}^{\text{ensemble}}$	**92.5**	83.5	79.0	88.7	99.1	63.6	63.4

A: arousal; V: valence

$\mathcal{M}_{target+\%source}^{single}$: the best performing model learnt from available target data with a certain percent of the generic data

$\mathcal{M}_{generic}^{ensemble}$: the ensemble classifier of multiple weak generic classifiers

In sum, the presented experimental results demonstrate the effectiveness of the fast-PADMA method in efficiently and effectively using the available data. The comparison with the state-of-the-art performances on the public datasets show the effectiveness of the geometric facial features and segment-level feature representation. The detailed investigation into target/source training data ratio demonstrates the improvement gain possible from fast-PADMA over the user-specific, generic, and their naïve hybrid counterparts. Finally, the comparison among models with different ensemble methods shows the effectiveness of the ensemble mechanism in fusing target and source data.

VI. CONCLUSIONS AND FUTURE WORK

This paper presents fast-PADMA, which is designed to facilitate the rapid modeling of a user-adaptive facial affect model by learning from both the source and target users. fast-PADMA includes an innovative segment-level feature representation, individual data alignment, and ensemble learning mechanism.

Compared to conventional transfer learning, fast-PADMA does not require the use of computational distribution estimation to measure the individual similarity. Instead, an ensemble of weak generic classifiers is used to learn the commonality from similar source data, while simultaneously accommodating individual differences. Experimental results show the effectiveness of the segment feature representation and the validity of the data alignment technique in supporting model aggregation for ensemble learning.

Our finding conduces to the human-computer interaction by making it possible to rapidly model the facial affect in a practical fashion. Since our method relies on the AMIL technique [14], it requires no expertise for annotation. Judging from the evaluations on 4 public datasets, the proposed approach presents a promising potential in the real-use affect-involved applications.

In future work, we propose to further investigate the aggregation of weak generic classifiers. Since the source data is

not distributed to the client side, and we also wish to minimize computational cost on the client side, fast-PADMA simply sets the number of weak generic classifiers as the number of source subjects. However, this is not the only way to build an ensemble of weak classifiers and we foresee there could be other interesting possibilities. It would also be interesting to investigate the relation between computational cost and performance improvement.

ACKNOWLEDGMENT

We sincerely thank the reviewers for their time and effort.

REFERENCES

[1] E. Sariyanidi, H. Gunes, and A. Cavallaro, "Automatic analysis of facial affect: A survey of registration, representation and recognition," *TPAMI IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8828, pp. 1–22, 2014.
[2] E. Sangineto, G. Zen, E. Ricci, and N. Sebe, "We are not All Equal: Personalizing Models for Facial Expression Analysis with Transductive Parameter Transfer," in *Proceedings of the ACM International Conference on Multimedia - MM '14*, 2014, pp. 357–366.
[3] W.-S. Chu, F. De La Torre, and J. F. Cohn, "Selective Transfer Machine for Personalized Facial Action Unit Detection," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3515–3522.
[4] G. Zen, E. Sangineto, and E. Ricci, "Unsupervised Domain Adaptation for Personalized Facial Emotion Recognition," in *ICMI*, 2014.
[5] P. Viola, J. C. Platt, and C. Zhang, "Multiple Instance Boosting for Object Detection," in *Advances in neural information processing systems*, 2006, p. 1417.
[6] K. Sikka, A. Dhall, and M. S. Bartlett, "Classification and weakly supervised pain localization using multiple segment representation," *Image Vis. Comput.*, vol. 32, no. 10, pp. 659–670, 2014.
[7] A. B. Ashraf, S. Lucey, J. F. Cohn, T. Chen, Z. Ambadar, K. M. Prkachin, and P. E. Solomon, "The painful face – Pain expression recognition using active appearance models," *Image Vis. Comput.*, vol. 27, pp. 1788–1796, 2009.
[8] Y. Chen, J. Bi, and J. Z. Wang, "MILES: Multiple-instance learning via embedded instance selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, pp. 1931–1947, 2006.
[9] Z. Fu, A. Robles-Kelly, and J. Zhou, "MILIS: Multiple instance learning with instance selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, pp. 958–977, 2011.
[10] Y. Xiao, B. Liu, Z. Hao, and L. Cao, "A similarity-based classification framework for multiple-instance learning.," *IEEE Trans. Cybern.*, vol. 44, no. 4, pp. 500–15, Apr. 2014.
[11] V. Cheplygina, D. M. J. Tax, and M. Loog, "Dissimilarity-based Ensembles for Multiple Instance Learning," *IEEE Trans. Neural Networks Learn. Syst.*, pp. 1–12, 2014.
[12] A. Ruiz, J. Van de Weijer, and X. Binefa, "Regularized Multi-Concept MIL for weakly-supervised facial behavior categorization," in *Proceedings of the British Machine Vision Conference*, 2014.
[13] X. Huang, J. Kortelainen, G. Zhao, X. Li, A. Moilanen, T. Sepplanen, and M. Pietiklainen, "Multi-modal Emotion Analysis from Facial Expressions and Electroencephalogram," *Comput. Vis. Image Underst.*, vol. 147, pp. 114–124, 2016.
[14] M. X. Huang, G. Ngai, K. A. Hua, and S. C. F. Chan, "Identifying User-Specific Facial Affects from Spontaneous Expressions with Minimal Annotation," *To Appear Trans. Affect. Comput.*
[15] R. el Kaliouby and P. Robinson, "Generalization of a Vision-Based Computational Model of Mind-Reading," in *ACII'05 Proceedings of the First international conference on Affective Computing and Intelligent Interaction*, 2005, pp. 582–589.
[16] G. Littlewort, M. S. Bartlett, I. Fasel, J. Susskind, and J. Movellan, "Dynamics of Facial Expression Extracted Automatically from Video," in *Proc. of Computer Vision and Pattern Recognition Workshop, 2004. CVPRW '04*, 2004, p. 80.
[17] P. Michel and R. El Kaliouby, "Real time facial expression recognition in video using support vector machines," in *Proceedings of the 5th international conference on Multimodal interfaces - ICMI '03*, 2003, p. 258.
[18] M. F. Valstar, M. Mehu, B. Jiang, M. Pantic, and K. Scherer, "Meta-Analysis of the First Facial Expression Recognition Challenge.," *IEEE Trans. Syst. Man. Cybern. B. Cybern.*, Jun. 2012.
[19] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, S. Chew, and I. Matthews, "Painful monitoring: Automatic pain monitoring using the UNBC-McMaster shoulder pain expression archive database," *Image Vis. Comput.*, vol. 30, pp. 197–205, 2012.
[20] J. Chen, X. Liu, P. Tu, and A. Aragones, "Learning person-specific models for facial expression and action unit recognition," *Pattern Recognit. Lett.*, vol. 34, no. 15, pp. 1964–1970, Nov. 2013.
[21] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22. pp. 1345–1359, 2010.
[22] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for Transfer Learning," *Proc. 24th Int. Conf. Mach. Learn. - ICML '07*, pp. 193–200, 2007.
[23] Y. Freund and R. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, pp. 119–139, 1995.
[24] X. Xiong and F. De la Torre, "Supervised Descent Method and Its Applications to Face Alignment," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 532–539.
[25] Jianbo Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 888–905, 2000.
[26] M. Soleymani, J. Lichtenauer, T. Pun, and M. Pantic, "A multimodal database for affect recognition and implicit tagging," *IEEE Trans. Affect. Comput.*, vol. 3, pp. 42–55, 2012.
[27] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and 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*, 2010, no. July, pp. 94–101.
[28] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO Algorithm for SVM Classifier Design," *Neural Comput.*, vol. 13, pp. 637–649, 2001.
[29] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn, "DISFA: A spontaneous facial action intensity database," *IEEE Trans. Affect. Comput.*, vol. 4, pp. 151–160, 2013.
[30] J. R. J. Fontaine, K. R. Scherer, E. B. Roesch, and P. C. Ellsworth, "The World of Emotions Is Not Two-Dimensional," *Psychol. Sci.*, vol. 18, pp. 1050–1057, 2007.
[31] K. Sikka, A. Dhall, and M. Bartlett, "Weakly supervised pain localization using multiple instance learning," in *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 2013, pp. 1–8.
[32] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, pp. 5–32, 2001.
[33] N. S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression," *Am. Stat.*, vol. 46, pp. 175–185, 1992.