Spontaneous Subtle Expression Recognition: Imbalanced Databases & Solutions ⋆

Anh Cat Le Ngo[1], Raphael Chung-Wei Phan[1], John See[2]

[1] Faculty of Engineering,
Multimedia University (MMU), Cyberjaya, Malaysia
lengoanhcat@gmail.com, raphael@mmu.edu.my
[2] Faculty of Computing & Informatics,
Multimedia University (MMU), Cyberjaya, Malaysia
johnsee@mmu.edu.my

Abstract. Facial expression analysis has been well studied in recent years; however, these mainly focus on domains of posed or clear facial expressions. Meanwhile, subtle/micro-expressions are rarely analyzed, due to three main difficulties: inter-class similarity (hardly discriminate facial expressions of two subtle emotional states from a person), intra-class dissimilarity (different facial morphology and behaviors of two subjects in one subtle emotion state), and imbalanced sample distribution for each class and subject. This paper aims to solve the last two problems by first employing preprocessing steps: facial registration, cropping and interpolation; and proposes a person-specific AdaBoost classifier with Selective Transfer Machine framework. While preprocessing techniques remove morphological facial differences, the proposed variant of AdaBoost deals with imbalanced characteristics of available subtle expression databases. Performance metrics obtained from experiments on the SMIC and CASME2 spontaneous subtle expression databases confirm that the proposed method improves classification of subtle emotions.

1 Introduction

Emotion recognition is an ability which human beings learn through observations of facial expressions. Recognizing normal expressions tends to be easy; however, recognizing subtle or micro-expressions proves to be more elusive for untrained eye. Recognizing subtle facial expressions is a difficult task because of their very brief durations ($1/3s$ to $1/25s$); moreover, they usually happen suddenly and involuntarily. Frank et al. [1] sets up psychological experiments to quantify how accurate untrained and trained people can recognize five different subtle emotions of unseen subjects. It reported that a naive group achieved 32% accuracy while even a trained group can only achieve 47% accuracy. While these small expressions are easily misinterpreted, psychological studies [2, 3] show that subtle expressions sometimes convey vital information such as a brief glimpse into concealed and suppressed feelings. Therefore, affective computer vision-based

⋆ This work was funded by TM under UbeAware project

recognition systems could improve emotion-related activities. For instance, police could have non-intrusive means for monitoring suspects' abnormal and subtle emotions, doctors could use the system for identifying patients' responses through their subtle expressions; mediators could understand whether their offers would satisfy other parties, etc. These scenarios would be unrealistic without extremely accurate systems which could even outperform highly trained human experts.

Like any other developments of machine learning systems, the first important step is the preparation of training and testing samples. Though posed or normal expression databases are popular and highly accessible, subtle expressions databases are not easy to build due to complex collecting procedures. Video samples need to be recorded spontaneously while subjects are told to conceal their emotions after viewing short video clips. Ground-truths and duration of video samples are identified by human-experts who have undergone intensive Micro Expression Training Tools (METT) courses [4]. Due to these high requirements, their availability are scarce. So far, there are only three publicly released datasets: SMIC [5], CASME1 [6], and CASME2 [7]. As CASME2 is the latest extension of CASME1 from the same group of researchers, utilization of CASME2 alone will be sufficiently thorough. While both CASME2 and SMIC are recommended databases for evaluating subtle expression recognition systems, their imbalanced nature of the data distribution across expression classes and subjects, would be a challenging ordeal for generic classifiers. This problem needs careful consideration in the development of any subtle expression recognition system.

This paper aims to analyze the imbalances of spontaneous micro-expression databases (CASME2 and SMIC) as well as to propose an effective and robust subtle expression recognition scheme. Section 2 shows statistical composition of CASME2 and SMIC across different classes and subjects, notably high skewness of the databases especially when leave-one-subject-out is the main cross-validation approach. Furthermore, variety in frame lengths of video samples, another imbalanced characteristic of the databases, directly affects feature-extraction stage. Section 3 proposes a robust system toward addressing those imbalances. At first preprocessing steps like facial cropping and registration in Subsection 3.1 are employed to remove morphological facial differences. Then, temporal interpolation (TIM) is utilized for equalizing frame lengths among samples in Subsection 3.2, and LBP-TOP extracts spatial-temporal texture features from these samples in Subsection 3.3. Finally, a person-specific AdaBoost filter, a combination of general adaBoost classifier and selective transfer machine (STM) framework, is introduced for solving imbalanced natures of spontaneous subtle expression databases in Subsections 3.4, 3.5, and 3.6. Experimental results are shown and discussed in Section 4 for CASME2 and SMIC databases to verify robustness and usefulness of the proposed solutions. Section 5 summarizes aims and achievements of this paper.

2 Statistical Study of CASME2 and SMIC

Before SMIC and CASME2 databases are chosen as main data of spontaneous subtle expression for training, testing and evaluating the recognition system, they have to be statistically studied throughly. The CASME2 database has 257 video samples with frame rate of 60 frame per seconds (*fps*) and collected from 26 subjects. The SMIC database has 16 subjects and 164 samples with frame rate of 100 *fps*. Besides these differences in number of subjects and frame rates, CASME2 and SMIC also differ in terms of the distribution of samples with respect to expression classes. The CASME2 database has five different subtle-expression classes: happiness, disgust, repression, surprise and tense, labeled from C1 to C5 accordingly. The distribution of video samples across these classes is rather non-uniform, and this is reflected by the number of samples for each expression shown in Table 1. Meanwhile, SMIC has only three classes of subtle emotions: positive, negative and surprise, of which labels given are S1, S2 and S3 respectively.

As leave-one-subject-out cross validation (LOSOCV), which take test samples from only single subject and use sample from the other subjects as training samples, is recommended by several facial macro-expression recognition works [8, 9], it is necessary to analyze how video samples are distributed according to both subjects and expression classes. Tables 2a and 2b show frequency of samples according to a particular subject and expression type in CASME2 and SMIC respectively. In Table 2, only a few highlighted subjects, e.g. Subject 17 in CASME2 and Subjects 03, 04, 11, etc. in SMIC have samples for all expression classes. The rest of the subjects have no samples in at least one expression class. Furthermore, the number of samples is not distributed equally among available classes as well, e.g. Subject 17 of CASME2 database has 14 samples in C2 class but only 1 sample in C4 class. By this observation, it is clear that samples are non-uniformly distributed to each class and subject.

Table 1: Distribution of samples in CASME2 & SMIC databases

CASME2			SMIC		
Emotion	Label	# samples	Emotion	Label	# samples
Happiness	C1	33	Positive	S1	51
Disgust	C2	60	Negative	S2	70
Repression	C3	25	Surprise	S3	43
Surprise	C4	27			
Tense	C5	102			

Distribution of databases also depends on whether video frame or video samples are considered as a basic unit. For examples, if each C# class has a video sample, each class occupies 20% of the database according to video samples. However, if the C1 sample has 60 frames, the rest has 10 frames each sample,

Table 2: Numbers of samples & frames according to subjects and emotions

(a) CASME2

	# samples					# frames				
	C1	**C2**	**C3**	**C4**	**C5**	**C1**	**C2**	**C3**	**C4**	**C5**
01	1	2			6	41	177			279
02	1		5	3	4	100		459	193	344
03		1		1	5		61		41	370
04		2			3		142			158
05	1	1		5	12	61	55		305	805
06	1	1		2	1	56	58		101	46
07		6			3		383			198
08			1		2			78		193
09	6		5		3	481		490		268
10					13					934
11		4			6		314			450
12	2	5		4	1	173	354		318	66
13	2				6	157				305
14	3				1	203				76
15	1			1	1	61			91	58
16	2		1		1	182		24		113
17	7	14	9	1	3	453	876	605	91	126
18					3					144
19	3	3		5	3	272	178		275	143
20		2			9		82			578
21			1		1			96		31
22			2					203		
23	1	4	3		4	95	303	232		254
24		3		1	4		181		36	162
25		3		2	2		153		154	145
26	2	9			5	171	602			328

(b) SMIC

	# samples			# frames		
	S1	**S2**	**S3**	**S1**	**S2**	**S3**
01	1	3	2	33	95	53
02	5		1	168		29
03	6	22	11	159	627	308
04	5	10	4	146	268	117
05	1	1		43	30	
06	2	2		66	75	
07						
08		9	4		313	138
09	3		1	108		42
10						
11	1	3	3	32	93	141
12	1		8	38		313
13			10			409
14	5	3	2	199	147	99
15	2	1	1	74	50	49
16						
17						
18	5	2		173	85	
19	1		1	50		35
20	5	14	3	174	433	115

the distribution according to video frames are 60% for C1, 10% for C2, C3, C4, C5. As CASME2 and SMIC samples are recorded at various frame rates and durations, each sample has different frame lengths. Therefore, distributions of video samples and frames are unnecessarily and rarely identical. Furthermore, note that each sample video clip is the result of clipping a long continuous video footage of each subject over time based on detected onset and offset points, i.e. when an expression is first spotted and when it is no longer observed, respectively. The number of times a subject could exhibit a particular expression would vary from across subjects and across types of expressions.

Figures 1 and 2 show normalized distributions of samples and frames for databases CASME2 and SMIC with respect to expression classes and a single

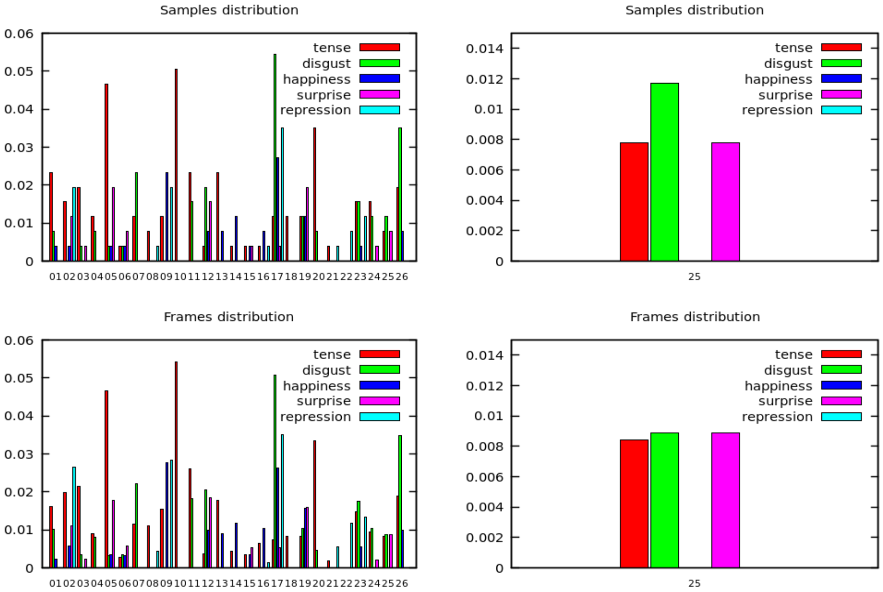

Fig. 1: CASME2 database analysis

selected subject. Subject indexes are shown in horizontal axes and there are a number of columns with various colors representing amount of samples available at each expression class of the subject. In general, the distributions of samples and frames in both databases are slightly different from each other. This discrepancy becomes more apparent when a single subject is considered; for example, Subject 25 of CASME2 in Figure 1 (left side) and Subject 11 of SMIC in Figure 2 (left side).

3 Subtle Expression Recognition

As unevenly distributed databases can cause significant problems to any machine learning system, this paper proposes robust techniques to tackle this imbalance in subtle expression recognition. As biases in the two spontaneous subtle expression databases were previously analyzed, this section focuses on describing the techniques used in our robust system to mitigate the imbalance. The solution follows the common 3-stage framework: preprocessing, feature extraction and classification. The first stage standardizes samples spatially by cropping faces according to eye positions using a Haar eye detector and registers faces of multiple subjects to a common facial model with fiducial points from Active Shape Model (ASM) [10]. Temporally, it also ensures the same number of frames are uniform for all samples by applying temporal interpolation model (TIM) [5]. In the second stage, Local Binary Pattern with Three Orthogonal Planes (LBP-TOP) [11], a

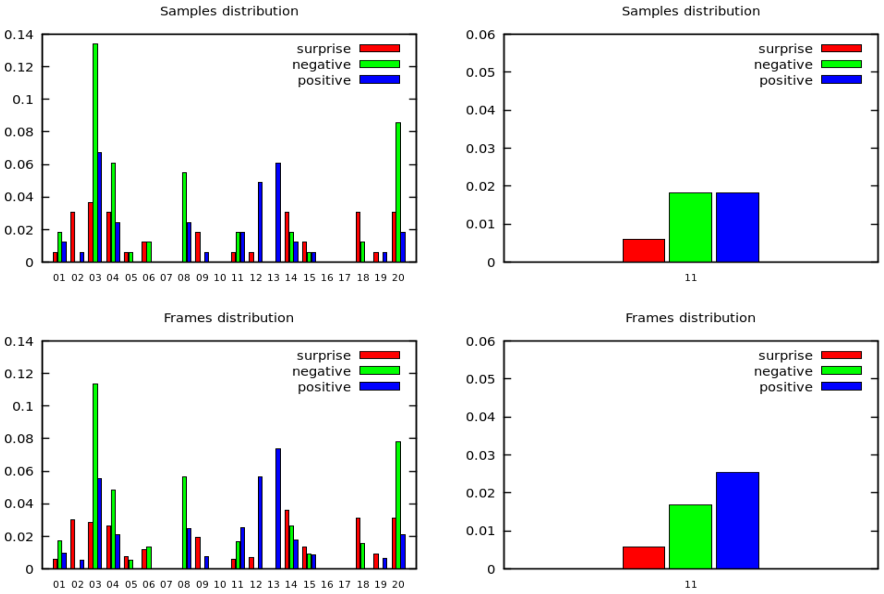

Fig. 2: SMIC database analysis

spatio-temporal local texture descriptor, is used to extract the main features for the learning and classification stage. As leave-one-subject-out cross-validation (LOSOCV) is adopted in the evaluation process, we propose a person-specific AdaBoost classifier with Selective Transfer Machine (STM) framework to deal with the person-specific bias and imbalanced training and testing datasets.

3.1 Face Cropping and Local Weighted Mean Transformation

As expressions are only caused by facial muscles, other background visual information is deemed to be irrelevant; therefore, it is filtered out from input data. In addition, facial structures of each subject are distinguished from each other, which is helpful in recognizing subjects' identity but obstructive towards generalizing classifiers for expressions (intra-class dissimilarity). Therefore, faces of multiple subjects need standardizing into a single common facial model M.

Let $S = [s_i | i = 1, \ldots, N]$ be a set of N subtle expression samples and each i^{th} sample be $s_i = [f_{i,j}, j = 1, \ldots, n_i]$, where n_i is the number of video frames for each i^{th} sequence s_i. A model face M is built from 68 facial landmark points $\psi(M)$, detected by ASM from a frontal face of a chosen subject in the database. Facial coordinates of the first frame $f_{i,1}$, i.e. $\psi(f_{i,1})$, are then used to estimate parameters of Local Weighted Mean (LWM) [12] transformation.

$$T = LWM(\psi(M), \psi(f_{i,i}))$$

This T can linearly transform faces from the rest of the sequence s_i according to the model face M. Finally, eye coordinates localized by a standard Haar feature-based cascaded eye detector are checked against the ASM coordinates and used to crop the transformed faces.

3.2 Temporal Interpolation Model

Video samples with different frame lengths may cause biases in the feature extraction and classification stages. Therefore, the temporal interpolating model (TIM) presented in [13, 5] is used for standardizing the number of frames in each sequence. This technique is able to produce the same number of frames for each sequence to reduce the bias effected upon the later stages. Previously, Zhou et al. [13] employed the TIM for synthesizing a talking mouth while Pfister et al. [5] applied it to micro-expression recognition to increase frame lengths before feature extraction. In this paper, we use TIM to balance the frame lengths across all samples in a database to provide temporal standardization. TIM operates on the assumption that frames of subtle expression samples form a continuous function, a curve in a low-dimensional manifold. In other words, a sequence of frames can be represented by a path graph P_n with n vertices, corresponding to n frames. Edges of P_n form an adjacency matrix $\mathbf{W} \in \{0, 1\}^{n,n}$ whereof $W_n = 1$ means direct connection between two vertices and $W_n = 0$ means otherwise. Mathematically, edges are defined as follows.

$$W_{i,j} = \begin{cases} 1, & \text{if } |i - j| = 1 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Parameters of manifolds, for n embedded vertices, can be found by mapping P_n to a line such that it minimizes distances between connected vertices.

$$\arg\min_{\mathbf{y}} \sum_{i,j} (y_i - y_j)^2 W_{i,j}, \quad i, j = 1, 2, \ldots, n \tag{2}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ are projections of video frames on the manifold of the graph path P_n. Obtaining $\mathbf{y}$ is equivalent to calculating the eigenvectors of the Laplacian graph of P_n such that it has eigenvectors $\mathbf{y_1}, \mathbf{y_2}, \ldots, \mathbf{y_{n-1}}$. Linear extension of graph embedding [14] allows finding linear projection w from zero-mean vectorized image x such that the objective function (2) is satisfied.

$$\arg\min_{w} \sum_{i,j} (\mathbf{w^T x_i} - \mathbf{w^T x_j})^2 W_{i,j}, \quad i, j = 1, 2, \ldots, n \tag{3}$$

He et al. [15] solves the resulting eigenvalue problem

$$\mathbf{XLX^T w} = \lambda^{'} \mathbf{XX^T w} \tag{4}$$

by using the singular value decomposition with $\mathbf{X} = \mathbf{U}\varSigma\mathbf{V^T}$. In [13], Zhou et al. show that the interpolated images can be computed as follows:

$$x = \mathbf{UM}\mathcal{F}^n(t) \tag{5}$$

where $\mathbf{M}$ is a square matrix and $\mathcal{F}^n(t)$ is a resulting curve of the Laplacian graph P_n. Let $\theta \in \{10, 15, 20\}$ be the number of interpolated frames for each video sample. The best value for θ is determined empirically by experiments for each evaluated database.

3.3 Local Binary Pattern with Three Orthogonal Planes

Local Binary Pattern (LBP) [16] is a popular texture operator that thresholds the local neighborhoods of each pixel in an image and converts them into a binary value. The binary values are then counted to form a histogram of different binary patterns. Zhao el al. [11] extended the LBP to LBP-TOP for use with dynamic spatio-temporal textures. LBP-TOP performs the classic LBP on all three orthogonal planes (XY, XY, YT) lying in a volumetric neighborhood. As a result, three sets of descriptors along the three orthogonal planes— LBP_{XY},LBP_{YZ}, and LBP_{XZ}, are then concatenated into a single histogram to extract the features for each sample clip. Owing to its robustness towards illumination and noise, LBP-TOP is our choice of feature extractor in our proposed scheme.

3.4 Selective Transfer Machine

In Subsection 3.2, the proposed system utilizes TIM before LBP-TOP feature extraction to solve imbalances of datasets caused by differences in frame lengths between video samples. This section focuses on imbalances caused by leave-one-subject-out cross-validation (LOSOCV) approach. It is due to infrequent distribution of training and testing video samples among subjects, as shown in Table 3a for the CASME2 corpus and Table 3b for the SMIC corpus and thoroughly described in Section 2. Though this imbalance is apparent, it is unavoidable due to practical difficulties in psychological experiments of collecting and evaluating spontaneous subtle emotion databases [6, 7, 17]. In these experiments, after stimuli are shown to subjects, their facial responses are recorded continuously in a long video, which is then post-processed into several short clips according to particular expressions exhibited at different moments. Subjects respond differently to the same stimuli; for instances, females express wider and more intense emotions than males do [18]. These differences would affect judgments of METT experts about types, numbers, on/off-set frames of subtle emotions samples.

As both released corpora of spontaneous subtle expressions suffer from biases while the existence of perfectly unbiased databases is highly improbable, it is necessary to develop domain adapted learning algorithms which can cope with such modern biased datasets [19]. Aytar and Zisserman [20] suggest utilization of pre-learned models in regularizing the training of new object class. Khosla et al [21] integrate a specific and a common discriminative model to remove biases. These techniques are supervised solutions requiring one or more labeled test samples in advance. This requirement is unrealistic for some subjects in CASME2 and SMIC corpora since there is only one sample for one expression. Moreover, it is infeasible to identify subtle emotions of unseen subjects without METT trained experts.

Table 3: Training & Testing Sample Distribution

(a) CASME2

	# samples									
	training					**testing**				
	C1	**C2**	**C3**	**C4**	**C5**	**C1**	**C2**	**C3**	**C4**	**C5**
01	32	58	27	25	96	1	2	0	0	6
02	32	60	22	22	98	1	0	5	3	4
03	33	59	27	24	97	0	1	0	1	5
04	33	58	27	25	99	0	2	0	0	3
05	32	59	27	20	90	1	1	0	5	12
06	32	59	27	23	101	1	1	0	2	1
07	33	54	27	25	99	0	6	0	0	3
08	33	60	26	25	100	0	0	1	0	2
09	27	60	22	25	99	6	0	5	0	3
10	33	60	27	25	89	0	0	0	0	13
11	33	56	27	25	96	0	4	0	0	6
12	31	55	27	21	101	2	5	0	4	1
13	31	60	27	25	96	2	0	0	0	6
14	30	60	27	25	101	3	0	0	0	1
15	32	60	27	24	101	1	0	0	1	1
16	31	60	26	25	101	2	0	1	0	1
17	26	46	18	24	99	7	14	9	1	3
18	33	60	27	25	99	0	0	0	0	3
19	30	57	27	20	99	3	3	0	5	3
20	33	58	27	25	93	0	2	0	0	9
21	33	60	26	25	101	0	0	1	0	1
22	33	60	25	25	102	0	0	2	0	0
23	32	56	24	25	98	1	4	3	0	4
24	33	57	27	24	98	0	3	0	1	4
25	33	57	27	23	100	0	3	0	2	2
26	31	51	27	25	97	2	9	0	0	5

(b) SMIC

	# samples					
	training			**testing**		
	S1	**S2**	**S3**	**S1**	**S2**	**S3**
01	49	67	42	2	3	1
02	50	70	38	1	0	5
03	40	48	37	11	22	6
04	47	60	38	4	10	5
05	51	69	42	0	1	1
06	51	68	41	0	2	2
07	51	70	43	0	0	0
08	47	61	43	4	9	0
09	50	70	40	1	0	3
10	51	70	43	0	0	0
11	48	67	42	3	3	1
12	43	70	42	8	0	1
13	41	70	43	10	0	0
14	49	67	38	2	3	5
15	50	69	41	1	1	2
16	51	70	43	0	0	0
17	51	70	43	0	0	0
18	51	68	38	0	2	5
19	50	70	42	1	0	1
20	48	56	38	3	14	5

Therefore, our proposed recognition system employs Selective Transfer Machine (STM), an unsupervised approach that re-samples weights for each training sample in order to fill up gaps or mismatches between distributions of training and testing samples. The STM framework jointly optimizes weights of training samples as well as losses of any classifiers; thus, preserving the discriminative property of decision boundaries on the re-weighted dataset. Furthermore, performances of any classifier trained on this dataset should improve since its model would more likely fit better to the testing dataset. As STM is a classifier-independent technique, there exists a single general formulation regardless of the choice of classifiers or classifiers' parameters. Let a training set be denoted as

$\mathcal{D}^{tr} = \{\mathbf{x}_i, y_i\}_{i=1}^{n^{tr}}, y_i \in \{+1, -1\}$, STM can be formulated as:

$$(\mathbf{w}, \mathbf{s}) = \arg\min_{\mathbf{w},\mathbf{s}} R_{\mathbf{w}}(\mathcal{D}^{tr}, \mathbf{s}) + \lambda \Omega_{\mathbf{s}}(\mathbf{X}^{tr}, \mathbf{X}^{te}) \tag{6}$$

where $R_{\mathbf{w}}(\mathcal{D}^{tr}, \mathbf{s})$ is the classifier loss on $\mathcal{D}^{tr}$ with vector of weights for each instance $\mathbf{s} \in \mathbb{R}^{n^{tr}}$, and learning coefficients $\mathbf{w}$. $\Omega_{\mathbf{s}}(\mathbf{X}^{tr}, \mathbf{X}^{te})$ measures dissimilarity between training and testing distribution. λ is a trade-off constant to balance the loss and distribution dissimilarity. As STM simultaneously optimizes a classifier loss and shifts a model such that it fits a subject's testing samples better, the final model can effectively remove biases caused by the person-specific bias.

3.5 AdaBoost Classifier

A boosted classifier is a linear combination of several weak classifiers in the form $\mathbf{H}(\mathbf{x}) = \sum_t w_t \mathbf{h}_t(\mathbf{x})$. It can be trained by greedily minimizing a loss function ϵ or optimizing scalar w_t and weaker learners $\mathbf{h_t}()$. Initially, a non-negative weight w_i derived from loss function ϵ is assigned to each sample $\mathbf{x_i}$ at the beginning. After each iteration, misclassified samples are weighted more heavily; thereby, losses of getting the same samples misclassified become severe in the next iterations. This is the basic principle of boosting algorithms (AdaBoost,LogitBoost, or L2Boost). They all need to iteratively classify samples given the sample weights.

In this paper, shallow trees are chosen as weak learners. The decision trees h_{TREE}, composed of a stump $h_j(\mathbf{x})$ at every non-leaf nodes j, are quickly trained in the manner proposed by Appel el al. [22]. Each stump generates a binary decision given an input $\mathbf{x} \in R^K$, polarity parameter $p \in \{\pm 1\}$, a threshold $\tau \in R$ and a feature index $k \in \{1, 2, \ldots, K\}$.

$$h_j(\mathbf{x}) = p_j \text{sign}(\mathbf{x}[k_j] - \tau_j) \tag{7}$$

Decision stump training can be used for classification if the goal at each stage is minimizing the weighted classification error ϵ.

$$\epsilon = \frac{1}{Z}\left[\sum_{\mathbf{x_i}[k] \le \tau} w_i \mathbf{1}_{\{y_i = +p\}} + \sum_{\mathbf{x_i}[k] > \tau} w_i \mathbf{1}_{\{y_i = -p\}}\right], \quad Z = \sum w_i \tag{8}$$

This error is minimized by selecting a single best feature k^* from all features K at each iteration. Determining the optimal threshold τ^* is costly due to $\mathcal{O}(N, K)$ accumulation of N weights, corresponding to N samples $\mathbf{x_i}[k]$, into discrete bins of feature values and indexes histogram. Appel el al. proves a bound on error of a decision stump given its preliminary errors on a subset of training data, which helps to identify and prune unpromising features early. In this paper, a fast AdaBoost training algorithm [22] is utilized to exploit this bound which reduces training time by an order of magnitude without any loss in performances.

3.6 Person-Specific AdaBoost Classifier

AdaBoost, a generic classifier, is regarded as a versatile algorithm in machine learning. However, it is not designed to accommodate person-specific bias, which often occurs in the LOSOCV approach. Moreover, AdaBoost neglects individual marked variety in facial morphology and behavior so it does not cope well with imbalanced data and generalize well to unseen faces well. Therefore, this paper proposes Person-specific AdaBoost classifier, which integrates AdaBoost with STM framework to transfer knowledge of testing samples onto distribution of training samples. Based on the generic classifier with STM defined by Eq. 6, the AdaBoost classifier with STM is formulated as,

$$(\mathbf{w}, \mathbf{s}) = \arg\min_{\mathbf{w},\mathbf{s}} \epsilon_{\mathbf{w}}(\mathcal{D}^{tr}, \mathbf{s}) + \lambda \Omega_{\mathbf{s}}(\mathbf{X}^{tr}, \mathbf{X}^{te}) \tag{9}$$

where $\epsilon_{\mathbf{w}}$ is the loss function of AdaBoost classifier (Eq. 8). To minimize Eq. 9 requires Alternate Convex Search method [23] since the STM objective function in Eq. 9 is biconvex, i.e. convex in $\mathbf{w}$ when $\mathbf{s}$ is fixed, and convex in $\mathbf{s}$ when $\mathbf{w}$ is fixed. A biconvex problem is guaranteed to converge with alternated optimization approach since its objective function monotonically converged to a critical point. The optimization process is shown in Algorithm 1.

Algorithm 1 AdaBoost with Selective Transfer Machine

```
Input: $\mathbf{X^{tr}}$, $\mathbf{X^{te}}$, number of weak classifiers N , $\lambda$
Output: instance-wise weights $\mathbf{w}$ of modified adaBoost and $\mathbf{s}$ of STM
  Initialize training loss $\epsilon_{\mathbf{w}} \leftarrow 0$;
  for i = 1:N do
      Find $\mathbf{s}$ of STM by solving the QP in Eq. 10.
      Find $\mathbf{w}$ of AdaBoost by solving the modified AdaBoost in Eq. 11
  end for
```

Fixing w and Minimizing over s: Denote training losses as $\epsilon_w = \epsilon_w(D^{tr}, \mathbf{s})$. The optimization over $\mathbf{s}$ $\Omega_s(X^{tr}, X^{te})$ in Eq. 6 can be re-written as a quadratic programming (QP) problem.

$$\min_{s} \frac{1}{2}\mathbf{s}^T \mathbf{K} \mathbf{s} + \left(\frac{C}{\lambda}\epsilon - \kappa\right) \tag{10}$$

$$\text{s.t. } 0 \leq s_i B, n_{tr}(1-\epsilon) \leq \sum_{i=1}^{n_{tr}} s_i \leq n_{tr}(1+\epsilon)$$

where $\mathbf{K}$ is a nonlinear kernel matrix of training samples, κ measures closeness between training and each test sample, and B defines upper bound of weights $\mathbf{s}$. This can be solved efficiently by interior point methods or Alternating Direction Method of Multipliers (ADMM) [24]. Further details on how this problem is optimized can be found in [25].

Fixing s and Minimizing over w: As STM is formulated regardless of the types of classifier, the objective function (Eq. 6) can take on a modified AdaBoost loss function (Eq. 8) instead of a general loss function $R_{\mathbf{w}}\left(\mathcal{D}^{tr}, \mathbf{s}\right)$:

$$\epsilon_s(\mathcal{D}^{tr}, \mathbf{w}) = \frac{1}{Z}\left[\sum_{\mathbf{x_i}[k]\leq\tau} s_i w_i \mathbf{1}_{\{y_i=+p\}} + \sum_{\mathbf{x_i}[k]>\tau} s_i w_i \mathbf{1}_{\{y_i=-p\}}\right], \quad Z = \sum s_i w_i \tag{11}$$

where s_i is a weight assigned to the i^{th} sample for matching distributions of training and testing data, and w_i is an instance-wise weight identifying risks of misclassifying the i^{th} sample. Beside additional weights s_i, the optimization processes strictly follow proposals by Appel el al. [22].

4 Experiments & Discussion

In Section 2 and Subsection 3.4, CASME2 and SMIC databases, two most comprehensive spontaneous subtle expressions databases, are statistically analyzed for their imbalance in a leave-one-subject-out cross-validation setting. Hence, these databases are suitable for evaluating the effectiveness of the proposed method in dealing with such imbalances. The experiments are set up with the following common parameters: number of interpolated frames for TIM and learning method AdaBoost with or without STM. Results demonstrate that rebalancing the number of frames in each sample by TIM interpolation can improve classifier performances and show the effectiveness of STM in personalizing a generic classifier AdaBoost while partly reducing person-specific bias during classification.

There are specific settings for CASME2 and SMIC due to different composition in each database. For instance, 5-class classification of subtle emotions (happiness, disgust, repression, tense, surprise) is performed on the CASME2 while SMIC is a 3-class classification problem (positive, negative and surprise). Furthermore, LBP-TOP extracts features from CASME2 and SMIC using different sets of parameters. Let us denote $R = (R_X, R_Y, R_T)$ as the radii of three orthogonal planes, and $S = (S_X, S_Y, S_T)$ as the number of block partitions used along the X, Y and T dimensions. More details on these LBP-TOP parameters can be found in [11]. In CASME2 database, LBP-TOP is used with the parameters $R = (1, 1, 4)$, $S = (5, 5, 1)$ while the LBP-TOP parameters for SMIC database are fixed to $R = (1, 1, 4)$, $S = (8, 8, 1)$. These values are suggested by the authors of each respective database [7, 5].

Besides the usage of commonly used parameters, fairness in evaluation also depends on the choice of performance metrics. For machine learning problems and classification tasks on highly skewed databases (like CASME2 and SMIC), the *accuracy* rate is an inadequate measure of the effectiveness of a classifier despite its popularity in literature. This is due to its susceptibility of inaccuracies due to heavily skewed data, i.e. unequal number of samples per class. The accuracy metric shows the average "hit rate" of all classes; therefore, it does

Table 4: Performance evaluation of subtle expression classification with leave-one-subject-out cross-validation

(a) CASME2 - 5-class recognition

No	***Learning Method***	***TIM***	***Accuracy***	***Precision***	***Recall***	***F_measure***
1	AdaBoost		0.3945	0.212	0.5095	0.2609
2	AdaBoost + STM		0.3876	0.2104	0.4824	0.2593
3	AdaBoost	TIM10	0.4015	0.2416	0.5242	0.2908
4	AdaBoost + STM	TIM10	0.4216	0.2587	0.5729	0.3077
5	AdaBoost	TIM15	0.422	0.2472	0.5455	0.3089
6	AdaBoost + STM	TIM15	**0.4378**	**0.291**	**0.532**	**0.3337**
7	AdaBoost	TIM20	0.365	0.2184	0.4414	0.2563
8	AdaBoost + STM	TIM20	0.3887	0.2257	0.4585	0.2672

(b) SMIC - 3-class recognition

No	***Learning Method***	***TIM***	***Accuracy***	***Precision***	***Recall***	***F_measure***
1	AdaBoost		0.4453	0.3424	0.6844	0.3994
2	AdaBoost + STM		0.4446	0.2947	0.6341	0.3611
3	AdaBoost	TIM10	0.4343	0.3706	0.7209	0.4356
4	AdaBoost + STM	TIM10	**0.4434**	**0.4009**	**0.7393**	**0.4731**
5	AdaBoost	TIM15	0.3677	0.2394	0.5392	0.2993
6	AdaBoost + STM	TIM15	0.3651	0.2955	0.7251	0.3703
7	AdaBoost	TIM20	0.3855	0.3244	0.6607	0.4036
8	AdaBoost + STM	TIM20	0.4285	0.3093	0.6937	0.3848

not reflect how well a machine learning method performs for each class. Therefore, additional metrics such as *precision*, *recall* and *f-measure* are necessary to provide a better measure of a classifier's performance [26].

For a multi-class classification task, confusion matrices proved to be more informative about behaviors of the evaluated classifier. The confusion tables are well-summarized by precision, recall and f-measure if a correctly retrieved positive data is the only important target. Each measure is the average of the same measures calculated for each class to evaluate unbalanced classes fairly regardless of the number of samples they have. In a binary or one-versus-all multi-class classification, precision is the number of true positive samples divided by the number of classified positive samples. Recall is the number of true positive samples divided by the number of ground-truth positive samples. F-measure is the overall combination of precision and recall which reflects relations between classified positive examples and ground-truth positive examples. As the leave-one-subject-out cross-validation (LOSOCV) approach is employed, evaluating the datasets produces N measurement sets where N is a number of subjects. Thus, the final value of each measure is an average of the same measures across N subjects so that performance of classifiers are fairly evaluated and independent from imbalances in the sample distribution across subjects. Tables 4a and 4b show the experimental results on CASME2 and SMIC databases. Experiments 1 and 2 of both tables contain results of subtle emotions classification without equalization of frame length in each video sample; while the remaining experiments are carried out with TIM## (TIM10,TIM15,TIM20), interpolating a video sample into a fixed number {10, 15, 20} of frames. Performances on both CASME2 and SMIC show a marked improvement with deployment of TIM; for instance, experiments 3-6 in Table 4a show better classifier performance on the CASME2 corpus, especially when TIM15 is used. Meanwhile, experiments 3-4 in Table 4b demonstrate that classifiers with TIM10 outperform those without TIM, giving the best performances among all experiments on the SMIC database. These results indicate the important role of TIM in rebalancing frame lengths of video samples across the entire corpus, reducing the effect of biases on the LBP-TOP feature extraction stage. Moreover, experimental results also highlight appropriate choices in the number of interpolated frames; for instance, TIM15 for the CASME2 corpus and TIM10 for the SMIC corpus shown in Tables 4a and 4b. Interestingly, over-interpolation does not help but harm the performance of classifiers due to an increase in artificial noises inherited from the interpolation process. The effects of that can be observed in experiments 7-8 of Table 4a and experiments 5-8 of Table 4b.

Furthermore, experimental results also proved the importance of utilizing the Selective Transfer Machine (STM) framework in reducing the effects of person-specific biases on the overall performance of AdaBoost classifiers used. Highlighted results in the Tables 4a and 4b demonstrate that AdaBoost classifiers with STM (on the best TIM setting discussed earlier) outperform those without STM in all four metrics, when all other parameters and conditions are identical. In general, superiority of classifiers with STM can be observed from the pairwise results of experiments 3-4 and 5-6 on both tables; while it appears to be less obvious in experiments 7-8. Again, the unprecedented results in experiments 7, 8 may be caused by a seemingly more prominent issue of TIM over-interpolation.

5 Conclusion

Imbalanced datasets are unavoidable practical problems in developing spontaneous subtle expression classification solution especially when a leave-one-subject-out cross-validation is the main evaluation approach. This paper proposes the use of TIM and STM to tackle the imbalances in the frames and sample levels. While TIM uses interpolation techniques to equalize frame lengths for all video samples, STM helps to personalize AdaBoost classifier, attenuate person-specific bias and reduce effects of imbalanced sample distribution in training and testing datasets. Experiments and performance validation are carefully designed to quantify the effectiveness of the proposed solution. Most importantly, the experimental results confirm that the solutions improve the overall classification performance for all evaluation metrics. Future work will include further comparisons with other solutions for imbalaned datasets e.g. TrAdaBoost [27].

References

1. Frank, M., Herbasz, M., Sinuk, K., Keller, A., Nolan, C.: I see how you feel: Training laypeople and professionals to recognize fleeting emotions. In: The Annual Meeting of the International Communication Association. Sheraton New York, New York City. (2009)
2. Ekman, P.: Lie catching and microexpressions. The philosophy of deception (2009) 118–133
3. Gottman, J.M., Levenson, R.W.: A two-factor model for predicting when a couple will divorce: Exploratory analyses using 14-year longitudinal data*. Family process **41** (2002) 83–96
4. Ekman, P.: Microexpression training tool (mett). San Francisco: University of California (2002)
5. Pfister, T., Li, X., Zhao, G., Pietikainen, M.: Recognising spontaneous facial micro-expressions. In: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE (2011) 1449–1456
6. Yan, W.J., Wu, Q., Liu, Y.J., Wang, S.J., Fu, X.: Casme database: a dataset of spontaneous micro-expressions collected from neutralized faces. In: Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on, IEEE (2013) 1–7
7. Yan, W.J., Li, X., Wang, S.J., Zhao, G., Liu, Y.J., Chen, Y.H., Fu, X.: Casme ii: An improved spontaneous micro-expression database and the baseline evaluation. PloS one **9** (2014) e86041
8. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on, IEEE (2010) 94–101
9. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: A comprehensive study. Image and Vision Computing **27** (2009) 803–816
10. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models-their training and application. Computer vision and image understanding **61** (1995) 38–59
11. Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. Pattern Analysis and Machine Intelligence, IEEE Transactions on **29** (2007) 915–928
12. Goshtasby, A.: Image registration by local approximation methods. Image and Vision Computing **6** (1988) 255–261
13. Zhou, Z., Zhao, G., Pietikainen, M.: Towards a practical lipreading system. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE (2011) 137–144
14. Yan, S., Xu, D., Zhang, B., Zhang, H.J., Yang, Q., Lin, S.: Graph embedding and extensions: a general framework for dimensionality reduction. Pattern Analysis and Machine Intelligence, IEEE Transactions on **29** (2007) 40–51
15. He, X., Cai, D., Yan, S., Zhang, H.J.: Neighborhood preserving embedding. In: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on. Volume 2., IEEE (2005) 1208–1213
16. Ojala, T., Pietikäinen, M., Mäenpää, T.: A generalized local binary pattern operator for multiresolution gray scale and rotation invariant texture classification. In: Advances in Pattern RecognitionICAPR 2001. Springer (2001) 399–408

17. Li, X., Pfister, T., Huang, X., Zhao, G., Pietikainen, M.: A spontaneous micro-expression database: Inducement, collection and baseline. In: Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on, IEEE (2013) 1–6
18. Brody, L.R.: On understanding gender differences in the expression of emotion. Human feelings: Explorations in affect development and meaning (1993) 87–121
19. Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE (2011) 1521–1528
20. Aytar, Y., Zisserman, A.: Tabula rasa: Model transfer for object category detection. In: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE (2011) 2252–2259
21. Khosla, A., Zhou, T., Malisiewicz, T., Efros, A.A., Torralba, A.: Undoing the damage of dataset bias. In: Computer Vision–ECCV 2012. Springer (2012) 158–171
22. Appel, R., Fuchs, T., Dollár, P., Perona, P.: Quickly boosting decision trees-pruning underachieving features early. In: JMLR Workshop and Conference Proceedings. Volume 28., JMLR (2013) 594–602
23. Gorski, J., Pfeuffer, F., Klamroth, K.: Biconvex sets and optimization with biconvex functions: a survey and extensions. Mathematical Methods of Operations Research **66** (2007) 373–407
24. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine Learning **3** (2011) 1–122
25. Chu, W.S., Torre, F.D.L., Cohn, J.F.: Selective transfer machine for personalized facial action unit detection. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE (2013) 3515–3522
26. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Information Processing & Management **45** (2009) 427–437
27. Dai, W., Yang, Q., Xue, G.R., Yu, Y.: Boosting for transfer learning. In: Proceedings of the 24th international conference on Machine learning, ACM (2007) 193–200