One-class Selective Transfer Machine
for Personalized Anomalous Facial Expression Detection

Hirofumi Fujita, Tetsu Matsukawa and Einoshin Suzuki

Graduate School/Faculty of Information Science and Electrical Engineering, Kyushu University, Fukuoka, Japan

Keywords: Anomalous Facial Expression, Personalization, One-class Classifier, Transfer Learning, Anomaly Detection.

Abstract: An anomalous facial expression is a facial expression which scarcely occurs in daily life and conveys cues about an anomalous physical or mental condition. In this paper, we propose a one-class transfer learning method for detecting the anomalous facial expressions. In facial expression detection, most articles propose generic models which predict the classes of the samples for all persons. However, people vary in facial morphology, e.g., thick versus thin eyebrows, and such individual differences often cause prediction errors. While a possible solution would be to learn a single-task classifier from samples of the target person only, it will often overfit due to the small sample size of the target person in real applications. To handle individual differences in anomaly detection, we extend Selective Transfer Machine (STM) (Chu et al., 2013), which learns a personalized multi-class classifier by re-weighting samples based on their proximity to the target samples. In contrast to related methods for personalized models on facial expressions, including STM, our method learns a one-class classifier which requires only one-class target and source samples, i.e., normal samples, and thus there is no need to collect anomalous samples which scarcely occur. Experiments on a public dataset show that our method outperforms generic and single-task models using one-class SVM, and a state-of-the-art multi-task learning method.

1 INTRODUCTION

Human interaction is carried out through not only verbal but also nonverbal communication such as facial expressions, gaze, gestures and body postures (Sangineto et al., 2014). Especially facial expressions provide cues about emotion, intention, alertness, pain and personality, regulate interpersonal behavior, and communicate psychiatric and biomedical status among other functions (Chu et al., 2013). An anomalous facial expression is defined, in this paper, as a facial expression which scarcely occurs in daily life. Such a facial expression conveys cues about an anomalous physical or mental condition. For example, a painful facial expression scarcely occurs in daily life, and conveys cues about an anomalous physical condition, e.g., pain. Detecting anomalous conditions is of crucial importance for human monitoring and in human-computer interaction.

In facial expression recognition, many of the crucial sources of error are individual differences in persons (Zeng et al., 2015). Age, gender and personality strongly influence the intensity and the way in which emotions are exhibited (Zeng et al., 2009). While a possible solution for handling individual differences would be to learn a single-task classifier from samples of the target person only, it will often overfit due to the small sample size of the target person in real applications. To handle these issues, several articles applied Transfer Learning (TL) methods which train personalized models from samples of the target and source persons (Chen et al., 2013; Chu et al., 2013; Sangineto et al., 2014; Chen and Liu, 2014; Mohammadian et al., 2016). Unlike single-task learning on only target samples, TL and Multi-task Learning (ML) models avoid overfitting using knowledge acquired from other domains or tasks.

Depending on the type of available labels on the source and target domains, samples for TL methods can be categorized into two types, multi-class or one-class samples. In the case when the multi-class samples are available for the source or target domain(s), several TL methods for facial expressions have been proposed (Chen et al., 2013; Chu et al., 2013; Sangineto et al., 2014; Chen and Liu, 2014; Mohammadian et al., 2016). Although they can classify facial expressions accurately using the information of their labels, collecting and annotating all class
samples are time-consuming. Especially in anomaly detection, it is difficult to collect anomalous samples because they scarcely occur. Therefore, for a wide range of applications including anomalous facial expression detection, it is important to develop a one-class TL method which can work with only one-class samples, i.e., normal samples. Since it is difficult to estimate an accurate boundary between the normal and anomalous samples from only one-class samples, developing a highly accurate one-class TL method is an open problem.

He et al. proposed a one-class ML method which requires only one-class samples in both the source and target domains (He et al., 2014). They conducted experiments on artificial toy data and textured images for detecting anomalous samples. Their approach detects anomalous samples by combining a generic model for all tasks and a single-task model. However, the generic model in (He et al., 2014) handles the source samples equally and thus does not handle the differences between the tasks. In fact, we found by experiments that the accuracy of their ML method is close to that of a conventional one-class method on the target person only.

To handle individual differences appropriately in one-class TL, we explore the idea of extending a generic model to a personalized model in a one-class classifier. Inspired by Selective Transfer Machine (STM) (Chu et al., 2013) which was proposed for multi-class TL, we propose a novel method named One-Class Selective Transfer Machine (OCSTM). OCSTM learns a personalized model from the one-class target and source samples by re-weighting the samples based on their proximity to the target samples. By handling the source samples unequally, OCSTM can handle the individual differences more appropriately than the conventional one-class ML method.

In summary, the main contributions of our work are as follows.

- We extend a multi-class selective transfer learning method (STM) to a one-class transfer learning method (OCSTM) for anomaly detection.
- We show the effectiveness of OCSTM compared to ordinary one-class methods and a one-class ML method by experiments on anomalous facial expression detection.
- Since the selection of feature extraction methods highly influences the performance of one-class methods, we compare them for anomalous facial expression detection by experiments.

2 RELATED WORK

In facial expression recognition, most articles focused on multi-class recognition which classifies face images into pre-defined classes, e.g., six basic expressions, namely happiness, sadness, anger, fear, surprise and disgust (Shan et al., 2009). Recognition of facial Action Units (AUs) (Ekman and Friesen, 1978), which represent changes in facial expression in terms of visibly observable movements of the facial muscles (Mohammadian et al., 2016), is also focused on analyzing information afforded by facial expression (Zeng et al., 2015). On the other hand, one-class facial expression classification, which distinguishes one-class facial expressions from the other ones, were reported in few articles. Zeng et al. proposed a method for distinguishing emotional facial expressions from non-emotional ones (Zeng et al., 2006). They formalized emotional facial expression detection as a one-class classification problem, and the classifier was learnt from emotional facial expressions of the target person only. The classifier discriminates the emotional facial expressions from the rest of the facial expressions. Since the emotional facial expression samples are often scarce in real applications, the one-class classifier learnt from emotional expressions of a single person may suffer from overfitting. Beyond a single-task method, He et al. proposed a ML method for one-class classification (He et al., 2014). However, as we explained in Sec. 1, their method handles the source samples equally and thus does not handle the differences between the tasks.

Chen and Liu proposed a TL method which uses binary-class (pain/normal) source samples and one-class (normal) target samples for pain recognition (Chen and Liu, 2014). They predicted the class distribution of the target person using the relationship between the class distributions of the persons in the source domain. Unlike (Chen and Liu, 2014), we propose a one-class transfer learning method which requires no anomalous samples in both the source and target domains.

In multi-class TL methods, several articles proposed to re-use source samples which are close to the target samples to handle individual differences. For instance, Chen et al. proposed a TL method which re-weights source samples so that distribution mismatch between the source and target domains is minimized (Chen et al., 2013). Chu et al. argued that re-weighting after predicting the densities is not practical and increases the estimation error (Chu et al., 2013). They proposed a TL method which re-weights the source samples without computing the source and target densities. In their method, a Support Vector Ma-
chine (SVM) based classifier is used for multi-class (binary-class) classification. Our approach is an extension of the method (Chu et al., 2013) to one-class classification based on One-Class Support Vector Machine (OCSVM) with non-linear kernel (Schölkopf et al., 2001). In (Sangineto et al., 2014; Zen et al., 2016), person-specific linear SVM classifiers for persons in the source domain were learnt, and knowledge about parameters of the classifiers was transformed to the target domain. However, since as we will see in Sec. 3.1 parameters of the separating hyperplane in nonlinear SVM are not computed explicitly, these methods cannot be directly applied to nonlinear OCSVM.

3 ONE-CLASS SELECTIVE TRANSFER MACHINE

In this section, we introduce our OCSTM for learning a personalized model from the target and source samples by re-weighting the samples based on their proximity to the target samples. Unlike multi-class transfer learning methods (Chen et al., 2013; Chu et al., 2013; Sangineto et al., 2014; Chen and Liu, 2014), OCSTM requires only one-class samples in both the source and target domains.

3.1 Overview of Our Method

Suppose we have the source samples \(X_{sc} = \{x_{sc}^i\}_{i=1}^{n_{sc}}\) and the target samples \(X_{tar} = \{x_{tar}^i\}_{i=1}^{n_{tar}}\), where \(x_{sc}^i, x_{tar}^i \in \mathbb{R}^d\), and \(n_{sc}\) and \(n_{tar}\) respectively represent the numbers of the samples of the source and target domains. Our goal is to learn a classifier \(f(x)\) which discriminates normal samples from anomalous samples in the target domain. The classifier \(f(\cdot)\) returns the value +1 if the input sample is predicted as normal, otherwise returns -1.

We use OCSVM (Schölkopf et al., 2001) because it is one of the most popular anomaly detection algorithms. The classifier of OCSVM is given by \(f(x) = \text{sign}(w^T \phi(x) - \rho)\), where \(\phi(\cdot)\) is the non-linear feature mapping associated with a kernel function \(k(x, y) = \phi(x)^T \phi(y)\), and \(w, \rho\) are parameters of a hyperplane. In most of the kernel functions such as Gaussian kernel, the mapped example \(\phi(x)\) cannot be calculated explicitly (Amari and Wu, 1999) and thus we cannot obtain the hyperplane parameter \(w\) explicitly. Instead, we obtain the inner products between the hyperplane parameter \(w\) and the mapped samples \(\phi(x)\) by the kernel function.

The objective function of OCSTM for learning the classifier is extended from that of STM (Chu et al., 2013) which uses SVM as the classifier. In STM, it is assumed that the labels of the target samples are not available. Thus, the target samples are used only for obtaining the weights for the source samples, and the classifier was learnt from the re-weighted source samples and their class labels. Since OCSVM is a one-class classifier, it requires no class label for learning. Therefore, the target samples can be used for classifier learning, as well as the re-weighted source samples. We formulate OCSTM as:

\[
(w, s) = \arg \min_{w, s} R_w(X_{sc}, X_{tar}, s) + \lambda \Omega_{sc}(X_{sc}, X_{tar}),
\]

where \(R_w(X_{sc}, X_{tar}, s)\) is the empirical risk (details are given in Sec. 3.1.1) defined on the source and target samples \(X_{sc}, X_{tar}\) with each instance \(x_{sc}\) and \(x_{tar}\) weighted by \(s_{sc} \in \mathbb{R}_{sc}\) and \(s_{tar} \in \mathbb{R}_{tar}\), respectively. Each element \(s_{sc}\) and \(s_{tar}\) corresponds to a non-negative weight for the sample \(x_{sc}\) and \(x_{tar}\), respectively. We denote \(s \in \mathbb{R}_{sc} \cup \mathbb{R}_{tar}\) as the vertical concatenation of \(s_{sc}\) and \(s_{tar}\) by \(s = (s_{sc}^1, \ldots, s_{sc}^{n_{sc}}, s_{tar}^1, \ldots, s_{tar}^{n_{tar}})^T\). The second term \(\Omega_{sc}(X_{sc}, X_{tar})\) measures the distribution discrepancy between the source and target distributions as a function of \(s\) (details are given in Sec. 3.1.2). The lower the value of \(\Omega_{sc}(X_{sc}, X_{tar})\), the more similar the source and target distributions are. A parameter \(\lambda (\geq 0)\) balances the empirical risk term and the distribution discrepancy term.

3.1.1 Empirical Risk

The first term in Eq. (1), \(R_w(X_{sc}, X_{tar}, s)\), is the empirical risk of OCSTM, where each instance is weighted by its proximity to the samples in the target domain. In OCSVM, the samples are mapped into the feature space associated with a kernel function, and are separated from the origin with maximum margin (Schölkopf et al., 2001). We introduce an error limit parameter \(v \in (0, 1]\), which, in OCSVM, corresponds to an upper bound on the fraction of anomaly samples on the source and target samples and a lower bound of the fraction of the support vectors (Schölkopf et al., 2001). We extend the objective function of OCSVM so that each training instance is weighted by \(s_i\) in the empirical risk of OCSTM. The empirical risk is defined by:

\[
R_w(X_{sc}, X_{tar}, s) = \frac{1}{2} ||w||^2 + \frac{1}{n_{all}} \sum_{i=1}^{n_{all}} s_i \xi_i - \rho,
\]

s.t. \(w^T \phi(x_i) \geq \rho - \xi_i, \xi_i \geq 0,\)

where \(n_{all} = n_{sc} + n_{tar}, \{x_i\}_{i=1}^{n_{all}} = X_{sc} \cup X_{tar}\), and \(\xi_i\) is a slack variable for training sample \(x_i\). If \(\xi_i\) is zero, the corresponding sample resides beyond the hyperplane,
otherwise below the hyperplane. In the second term, small weight \( s_i \) is given to the slack variable \( \xi_i \) of the sample which is far from the target samples and thus such a sample is hardly considered.

### 3.1.2 Domain Discrepancy

The second term in Eq. (1), \( \Omega_{sc}(X^sc, X^{tar}) \), is the domain discrepancy, which is used to find a re-weighting function for minimizing the discrepancy between the source and target domains. Following (Chu et al., 2013), we adopt the Kernel Mean Matching (KMM) (Gretton et al., 2009) to minimize the discrepancy between the means of the source and target distributions in the Reproducing Kernel Hilbert Space (RKHS) \( \mathcal{H} \). The difference between STM and OCSTM in the domain discrepancy is just the notation of the symbols\(^1\). KMM computes the instance-wise weights \( s^{sc} \) that minimizes

\[
\Omega_{sc}(X^{sc}, X^{tar}) = \left\| \frac{1}{n_{sc}} \sum_{i=1}^{n_{sc}} s^{sc}_i \phi(x^{sc}_i) - \frac{1}{n_{tar}} \sum_{j=1}^{n_{tar}} \phi(x^{tar}_j) \right\|^2_{\mathcal{H}},
\]

where \( \| \cdot \|_2 \) is L2-norm in RKHS. We introduce \( \kappa_i := \frac{n_{tar}}{n_{sc}} \sum_{j=1}^{n_{tar}} k(x^{sc}_i, x^{tar}_j), \) \( i = 1, \ldots, n_{sc} \), which captures the proximity between the source and each target sample in \( \mathcal{H} \), and two constraints \( s^{sc}_i \in [0, B], \) \( \frac{1}{n_{sc}} \sum_{i=1}^{n_{sc}} s^{sc}_i = 1 \leq \varepsilon. B \) in the former constraint limits the scope of the discrepancy between the source and target distributions and guarantees robustness by limiting the influence of each sample \( x^{sc}_i \). For \( B \to 1 \), we obtain the unwighted solution. \( \varepsilon \) in the latter constraint is for guaranteeing that the weighted source distribution is close to a probability distribution (Gretton et al., 2009). As in (Chu et al., 2013), the problem of finding suitable weights \( s^{sc} \) in Eq. (3) can be written as a quadratic programming (QP):

\[
\min_{s^{sc}} \frac{1}{2} (s^{sc})^T K^{sc} s^{sc} - \nu^T s^{sc},
\]

s.t. \( s^{sc}_i \in [0, B], \sum_{i=1}^{n_{sc}} s^{sc}_i = n_{sc}, \leq n_{sc} \varepsilon, \)

where \( K^{sc}_ij := k(x^{sc}_i, x^{sc}_j), \) \( i, j = 1, \ldots, n_{sc} \) and \( \kappa = (\kappa_1, \ldots, \kappa_{n_{sc}})^T \). A large value of \( \kappa_i \) indicates large importance of \( x^{sc}_i \) and is likely to lead to large \( s^{sc}_i \).

### 3.2 Optimization

To minimize the objective function in Eq. (1), we adopt the Alternate Convex Search (ACS) method\(^2\) (Gorski et al., 2007), which solves alternately two convex subproblems over hyperplane parameter \( w \) and selective instance-wise weights \( s^{sc} \). We assume that all target samples are equally important and thus we fix \( s^{tar} \) to a constant value. In this case, the objective function in Eq. (1) is biconvex, i.e., it is convex in \( w \) when \( s^{sc} \) is fixed, and is convex in \( s^{sc} \) when \( w \) is fixed. Under these conditions, the ACS approach is guaranteed to monotonically decrease the objective function.

Note that the way of optimizing \( w \) is different from that of STM. STM trains a nonlinear SVM in the primal problem using the representer theorem\(^2\) (Chapelle, 2007) due to its simplicity and efficiency. However, since the empirical risk of OCSTM also contains the hyperplane parameter \( \rho \), OCSTM cannot train OCSVM in the primal. Therefore, OCSTM trains OCSVM in the dual problem with Lagrange multiplier. In the following sections, we show that the subproblems are convex and how we optimize the subproblems.

#### 3.2.1 Optimization on \( w \)

When \( s \) is fixed, the subproblem over \( w \) corresponds to the minimization of the empirical risk \( \kappa_w(X^{sc}, X^{tar}, s) \) because \( \Omega_{sc}(X^{sc}, X^{tar}) \) does not depend on \( w \). Eq. (2) can be minimized with Lagrange multipliers \( \alpha, \beta \geq 0 \). The Lagrangian of Eq. (2) is given by:

\[
L(w, \xi, \rho, \alpha, \beta) = \frac{1}{2} \| w \|^2 + \frac{1}{n_{all}} \sum_{i=1}^{n_{all}} s_i \xi_i - \rho \sum_{i=1}^{n_{all}} \alpha_i (w^T \phi(x^i) - \rho + \xi_i) - \sum_{i=1}^{n_{all}} \beta_i \xi_i.
\]

The partial derivatives of the Lagrangian are set to zero, which leads to the following equations:

\[
w_i = \sum_{i=1}^{n_{all}} \alpha_i \phi(x^i),
\]

\[
\alpha_i = \frac{s_i}{\sqrt{n_{all}}} - \beta_i,
\]

\[
\sum_{i=1}^{n_{all}} \alpha_i = 1.
\]
The dual problem is obtained from Eqs. (5) and (6):
\[
\min \frac{1}{2} \alpha^T K_{\text{all}} \alpha, \quad \text{s.t. } 0 \leq \alpha_i \leq \frac{s_i}{\nu_{\text{all}}}, \sum_{i=1}^{n_{\text{all}}} \alpha_i = 1, \quad (7)
\]
where \( K_{ij} := k(x_i, x_j), i, j = 1, \ldots, n_{\text{all}} \). The subproblem is convex because \( K_{\text{all}} \geq 0 \). For any sample \( x_i \) whose corresponding \( \alpha_i \) and \( \beta_i \) are nonzero at the optimum, i.e., \( 0 < \alpha_i < s_i / (\nu_{\text{all}}) \), two inequality constraints in Eq. (2) become equalities, i.e., \( w^T \phi(x_i) - \rho + \xi_i = 0 \) and \( \xi_i = 0 \). From any such \( x_i \), we can recover \( \rho \) by the following equation:
\[
\rho = w^T \phi(x_i) = \sum_{j=1}^{n_{\text{all}}} \alpha_j k(x_j, x_i). \quad (8)
\]

We also recover the slack variables \( \xi = (\xi_1, \ldots, \xi_{n_{\text{sc}}}, \xi_{\text{tar}}) \) in Eq. (2) by considering two cases of \( \alpha_i \). If \( \alpha_i = 0 \) and \( \beta_i \neq 0 \), the second inequality constraint in Eq. (2) becomes equality, i.e., \( \xi_i = 0 \). Therefore, the first inequality constraint in Eq. (2) becomes \( w^T \phi(x_i) \geq \rho \). If \( 0 < \alpha_i \leq s_i / (\nu_{\text{all}}) \), the first inequality constraint in Eq. (2) becomes equality, i.e., \( \xi_i = \rho - w^T \phi(x_i) \). Finally, we can obtain \( \xi \) by the following equation, \( \xi_i = \max(0, \rho - w^T \phi(x_i)) \), \( i = 1, \ldots, n_{\text{all}} \). Using \( w \) in Eq. (6), the classifier of OCSTM is obtained as follows:
\[
f(x) = \text{sign}(w^T \phi(x) - \rho) = \text{sign} \left( \sum_{j=1}^{n_{\text{all}}} \alpha_j k(x, x_j) - \rho \right). \quad (9)
\]

### 3.2.2 Optimization on \( s_{\text{sc}} \)

When \( w \) is fixed, we obtain the subproblem over \( s_{\text{sc}} \) from Eqs. (2) and (4), which corresponds to the following QP:
\[
\min \frac{1}{2} (s_{\text{sc}})^T K_{\text{sc}} s_{\text{sc}} + \left( \frac{1}{\lambda n_{\text{all}}} \xi_{\text{sc}} - K \right)^T s_{\text{sc}},
\]
\[
\text{s.t. } 0 \leq s_i \leq B, \ n_{\text{sc}}(1 - \epsilon) \leq \sum_{i=1}^{n_{\text{sc}}} s_i \leq n_{\text{sc}}(1 + \epsilon), \quad (10)
\]
where \( \xi_{\text{sc}} = (\xi_{1}, \ldots, \xi_{n_{\text{sc}}}) \). The subproblem is convex because \( K_{\text{sc}} \geq 0 \). As in STM (Chu et al., 2013), the procedure here is different from the original KMM. In each iteration, the weights will be refined through the slack variables \( \xi_{\text{sc}} \). The source samples which have large \( \xi_{\text{sc}} \) lead to small \( s_{\text{sc}} \) to keep the objective small, hence this difference reduces the weights for samples which are close to anomalous samples. Different from STM, the slack variable of a source sample is computed without the class label. Therefore, the discriminative property between the normal and anomalous samples would highly depend on feature extraction methods, which we will address in Sec.4.2.

#### Algorithm 1: One-class selective transfer machine

**Input:** Samples \( X_{\text{sc}}, X_{\text{tar}} \), parameters \( \sigma, v, \lambda, B, \epsilon \)

**Output:** Hyperplane parameters \( w, \rho \) and instance-wise weights \( s \)

**Initialize** \( \xi \leftarrow 0 \)

**while** not converged **do**

- Obtain the instance-wise weights \( s_{\text{sc}} \) by solving the QP in Eq. (10)

**if** first loop **then**

- \( s_{\text{tar}} \leftarrow \max(s_{\text{sc}}) \mathbf{1} \)

**end if**

- Obtain the hyperplane parameters \( w, \rho \) by solving Eqs. (7) and (8)

**end while**

Algorithm 1 summarizes the OCSTM algorithm. While the instance-wise weights \( s_{\text{sc}} \) for the samples in the source domain are given in Eq. (10), such weights are not given to the samples in the target domain. When the classifier is learnt from the source and target samples, target samples need to be given instance-wise weights \( s_{\text{tar}} \). To give target samples large instance-wise weights, the elements of \( s_{\text{sc}} \) are set to the maximum value of \( s_{\text{sc}} \) after the first optimization of \( s_{\text{sc}} \).

## 4 EXPERIMENTS

We conducted four kinds of experiments for evaluating the proposed OCSTM regarding the following aspects: comparison with related methods, dependency on the feature extraction methods, performance with respect to the number of the training samples in the target domain, and dependency on the parameter \( \lambda \).

#### 4.1 Dataset

The UNBC-McMaster Shoulder Pain Expression Archive (UNBC-MSPEA) database (Lucey et al., 2011) is composed of 200 video sequences containing spontaneous pain facial expressions. It depicts

\(^{3}\) We denote the bandwidth of the Gaussian kernel by \( \sigma \), a zero-value vector whose length is \( n_{\text{all}} \) by \( \mathbf{0} \), a one-value vector whose length is \( n_{\text{sc}} \) by \( \mathbf{1} \), and the function that finds the maximum element of an input vector by \( \max(\cdot) \).
4.2 Feature Extraction

Since OCSTM is a one-class method, the following requirements for features are necessary: (1) normal and anomalous samples in the target domain are separated in the feature space, (2) there are at least a few source samples which are close to the target normal samples. If (1) is not satisfied, the algorithm of OCSTM gives large weights to the samples which are close to the anomalous samples. If (2) is not satisfied, all source samples are far from the target samples, and the source samples do not help to predict the classes of the target samples.

To investigate features suitable for the one-class methods, we applied two ordinary appearance-based extraction methods, Scale-Invariant Feature Transforms (SIFTDs) (Sangineto et al., 2014) and Local Binary Pattern Histograms feature (LBPH) (Ahonen et al., 2006) as well as simple landmark-based features, i.e., distances between pairs of landmarks (Fig. 1). For detecting face and facial points in the three feature extraction methods and for extracting the landmark-based features, AMM landmarks annotated to all images were employed.

SIFTD is a local feature, and is thus suitable for anomalous facial expressions analysis. This is because an anomalous facial expression is related to AUs which are localized to specific face regions. Firstly the face was detected, aligned, and resized to a $200 \times 200$ pixel window. Then descriptors were computed within $36 \times 36$ pixel regions around predetermined 16 facial landmarks (Fig. 1). The length of the descriptor is 128 for each region and thus the length of the feature is $128 \times 16 = 2,048$ for each image.

LBPH is also a local feature and thus suitable for anomalous facial expression analysis. Firstly, the face was detected, aligned, and resized to a $128 \times 128$ pixel window. Then the resized face image was divided to $8 \times 8$ blocks and the LBP histograms were extracted from the blocks. We apply $uniformLBP_{8,1}^{2}$ to each block, where $u_2$ means "uniform", and $(8,1)$ represents 8 sampling points on a circle of radius 1. From each block, a 59-dimensional feature was extracted and thus the length of the LBP histogram is $59 \times 8 \times 8 = 3,776$.

Face landmarks is one of the most applied features for facial expression analyses. Typically, high-dimensional features contain more noise than low-dimensional ones. Thus we use simple features, i.e., landmark distances, which are related to painful facial expressions. In (Lucey et al., 2011), PSPI was decided based on AUs, brow lowering (AU4), cheek raising (AU6), eyelid tightening (AU7), nose wrinkling (AU9), upper-lip raising (AU10) and eye closing (AU43). We selected seven distances related to the AUs in Table 1. The distances were computed based on normalized coordinates. Here x and y coordinates are respectively normalized by the distance between the inner corners of the eyes and the distance between the middle of the eyes and the nasal spine for each image. We refer to these Distances of Face Landmarks as DFL and the length of the feature is 7.

4.3 Evaluation Protocol and Parameter Setting

Since images in which a painful facial expression is exposed are scarce (1.97%), we used them as anomalous samples and normal samples as the rest. Following other articles, experiments were conducted using...
Table 2: Comparison with relevant methods on DFL features (ntar = 500). Each row shows the result when the each subject was used for the test subject and the last row shows their average.

<table>
<thead>
<tr>
<th>subject</th>
<th>F1 score</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I-OC SVM</td>
<td>ST-OC SVM</td>
</tr>
<tr>
<td>1</td>
<td>0.11</td>
<td>0.27</td>
</tr>
<tr>
<td>2</td>
<td>0.31</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0.88</td>
<td>0.93</td>
</tr>
<tr>
<td>4</td>
<td>0.60</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0.15</td>
<td>0.23</td>
</tr>
<tr>
<td>7</td>
<td>0.34</td>
<td>0.67</td>
</tr>
<tr>
<td>8</td>
<td>0.75</td>
<td>0.49</td>
</tr>
<tr>
<td>9</td>
<td>0.67</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>average</td>
<td>0.38</td>
<td>0.26</td>
</tr>
</tbody>
</table>

4In this case, Precision and Recall are both zero.

a leave-one-subject-out evaluation scheme in which one subject in turn was chosen as the target and the others as the source. Each image was treated independently, i.e., no temporal information was used. The ntar target samples were randomly selected from the target subject, and the rest of the target samples were used for the test. The Area Under the ROC Curve (AUC) and F1 score were used for evaluation, where $F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$. We define $F1 = 0$ when the number of anomalous samples which the classifier predicts as anomalous is zero. We repeated the experiments five times (except in Sec. 4.7 one time) on each person for each method, and the averages of AUC and F1 scores were reported.

We used Gaussian kernel with a bandwidth equal to the mean distance between the target training samples. We set the parameter ν, which implies an upper bound on the fraction of anomaly samples on the source and target samples, as ν = 0.0001 since we used only normal samples for training OCSTM.

Furthermore, we set three parameters differently from STM. Firstly, we set ε as follows. The first and second constraints in Eq. (7) together derive the following requirement for s, $1 \leq \frac{1}{\sum_{i=1}^{ntar} s_i}$. To ensure this inequality through the second constraint in Eq. (10), we set $\varepsilon = 1 - \frac{n_{all}}{n_{sc}}$.

Secondly, we set the parameter B, i.e., the upper bound of $s_i$ in Eq. (10), based on the ratio of the source samples. In (Chu et al., 2013), B was set to a large value. We observed that under such a setting, only a small number of source samples tend to be re-weighted largely, even if there are more source samples which are close to target samples. Therefore, we set B to the reciprocal of the ratio of the source samples which are close to target samples, i.e., $B = n_{sc}/n_{cl}$, where ncl is the number of source samples whose average similarity measured by the Gaussian kernel function to the target samples is larger than that between target samples. If nall = 0, we set $B = 10,000$ so that none of the $s_i$ reaches the upper bound B, and in this case $s_i$ does not depend on B.

Thirdly, we scale the parameter λ by nsc to balance the slack variables $\xi_i$ and $\kappa_i$. In Eq. (10), the $\frac{1}{n_{all}} \cdot \xi_i$ tends to be significantly smaller than $\kappa_i$ due to the term $\frac{1}{n_{all}} \cdot \kappa_i$. In addition, $\kappa_i$ is weighted by nsc by definition, i.e., $\kappa_i := \frac{n_{sc}}{n_{all}} \cdot \sum_{j=1}^{ntar} k(x_i^s, x_j^t)$. Since nsc and nall for the target person are different from those of the other persons, and nsc is nearly equal to nall, we set $\lambda := \frac{\lambda'}{n_{sc}}$. In Sec. 4.7, we investigated the dependency of OCSTM on the parameter $\lambda'$. Since the dependency is small, we set $\lambda' = 1,000$ in all experiments except in Sec. 4.7.

4.4 Comparison with Related Methods

In this section, we demonstrate the effectiveness of OCSTM for anomalous facial expression detection compared with related methods. Since we suppose that only one-class samples are available, we treat three one-class methods as compared methods, i.e., OCSVM trained on the only target samples (T-OCSVM), OCSVM trained on the source and target samples (ST-OCSVM), and a state-of-the-art multi-task learning with one-class SVM (ML-OCSVM) (He et al., 2014). These methods were implemented by us and the parameters for each method were tuned so that it exhibits the best result. For this comparison, we used DFL features.

Table 2 shows the F1 scores and AUC of the compared methods. We see that our approach outperforms
(a) T-OCSVM on target samples
(b) OCSTM on target and source samples

Table 3: Average similarities between the normal and anomalous samples in the target domain, computed by the Gaussian kernel function.

<table>
<thead>
<tr>
<th>subject</th>
<th>normal-anomalous</th>
<th>normal-normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.140</td>
<td>0.331</td>
</tr>
<tr>
<td>2</td>
<td>0.019</td>
<td>0.358</td>
</tr>
<tr>
<td>3</td>
<td>0.007</td>
<td>0.345</td>
</tr>
<tr>
<td>4</td>
<td>0.256</td>
<td>0.372</td>
</tr>
<tr>
<td>5</td>
<td>0.411</td>
<td>0.333</td>
</tr>
<tr>
<td>6</td>
<td>0.012</td>
<td>0.316</td>
</tr>
<tr>
<td>7</td>
<td>0.003</td>
<td>0.361</td>
</tr>
<tr>
<td>8</td>
<td>0.169</td>
<td>0.321</td>
</tr>
<tr>
<td>9</td>
<td>0.178</td>
<td>0.406</td>
</tr>
<tr>
<td>10</td>
<td>0.418</td>
<td>0.362</td>
</tr>
</tbody>
</table>

Fig. 2 shows 2D PCA projections of the samples and the hyperplanes for T-OCSVM and OCSTM when $n_{tar} = 10$. Green and red squares respectively represent normal and anomalous test samples in the target domain, and orange crosses and gray squares respectively represent normal training samples in the target and source domains. A red closed surface represents a hyperplane of each classifier which predicts a sample inside as normal.

4.5 Comparison of Feature Extraction Methods for OCSTM

In this section, we investigate the dependency of the proposed method on the feature extraction methods. As mentioned in Sec. 4.2, the two requirements for features are necessary in OCSTM. We conducted experiments using the three feature extraction methods.
Table 4: F1 score using three feature extraction methods for OCSVM and OCSTM ($n_{\text{tar}} = 500$).

<table>
<thead>
<tr>
<th>subject</th>
<th>LBPH</th>
<th>SIFTD</th>
<th>DFL</th>
<th>LBPH</th>
<th>SIFTD</th>
<th>DFL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.03</td>
<td>0.03</td>
<td>0.11</td>
<td>0.07</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>2</td>
<td>0.11</td>
<td>0.13</td>
<td>0.31</td>
<td>0.20</td>
<td>0.18</td>
<td>0.54</td>
</tr>
<tr>
<td>3</td>
<td>0.64</td>
<td>0.69</td>
<td>0.88</td>
<td>0.72</td>
<td>0.75</td>
<td>0.90</td>
</tr>
<tr>
<td>4</td>
<td>0.45</td>
<td>0.52</td>
<td>0.60</td>
<td>0.59</td>
<td>0.59</td>
<td>0.61</td>
</tr>
<tr>
<td>5</td>
<td>0.06</td>
<td>0.08</td>
<td>0</td>
<td>0.09</td>
<td>0.15</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0.06</td>
<td>0.06</td>
<td>0.15</td>
<td>0.08</td>
<td>0.08</td>
<td>0.21</td>
</tr>
<tr>
<td>7</td>
<td>0.11</td>
<td>0.14</td>
<td>0.34</td>
<td>0.18</td>
<td>0.18</td>
<td>0.56</td>
</tr>
<tr>
<td>8</td>
<td>0.52</td>
<td>0.59</td>
<td>0.75</td>
<td>0.60</td>
<td>0.62</td>
<td>0.73</td>
</tr>
<tr>
<td>9</td>
<td>0.35</td>
<td>0.41</td>
<td>0.67</td>
<td>0.44</td>
<td>0.43</td>
<td>0.82</td>
</tr>
<tr>
<td>10</td>
<td>0.46</td>
<td>0.45</td>
<td>0</td>
<td>0.63</td>
<td>0.52</td>
<td>0</td>
</tr>
</tbody>
</table>

| average | 0.28 | 0.31 | 0.38| 0.36 | 0.35 | 0.46|

Table 4 shows F1 scores of T-OCSVM and OCSTM using three kinds of features, LBPH, SIFTD and DFL. The OCSTM outperforms T-OCSVM for each feature extraction method in F1 score, and the results show that the source samples which are close to the target samples help to predict the classes of the test samples accurately. Table 4 also shows that the F1 scores using DFL are higher than those using features LBPH and SIFTD. This is because high-dimensional features, i.e., LBPH and SIFTD, contain more irrelevant information than low-dimension ones, i.e., DFL.

As mentioned in Sec. 4.4, the F1 scores of OCSTM with DFL features for subjects #5 and #10 are zero because the similarity between the normal and anomalous samples in the target domain is larger than that between the target normal samples. On the other hand, in OCSTM with LBPH and SIFTD features, we confirmed that the similarity between the normal and anomalous samples in the target domain is smaller than that between the target normal samples for all subjects. Thus the requirement (1) in Sec. 4.2 is satisfied and the F1 scores are not zero.

### 4.6 Performance Analysis in Terms of the Number of the Target Samples

In this section, we analyze how the performance of our method depends on the number of target samples $n_{\text{tar}}$. We conducted experiments using DFL by varying $n_{\text{tar}}$ from 10 to 500.

Fig. 3 shows the F1 scores and AUC of OCSTM and the compared methods. We see that the performance decreases as $n_{\text{tar}}$ decreases. OCSTM outperforms the other methods for each $n_{\text{tar}}$ in F1 score and AUC, except for AUC when $n_{\text{tar}} = 10$. The F1 score of OCSTM when $n_{\text{tar}} = 300$ is higher than those of the other methods when $n_{\text{tar}} = 500$. We can safely conclude that OCSTM avoids overfitting better than the other methods.

### 4.7 Dependency of OCSTM on the Parameter $\lambda$

Here we analyze how the performance of our method depends on the parameter $\lambda = \lambda' / n_{\text{tar}}^2$. We conducted experiments using DFL by varying $\lambda' \in \{1, 10, 100, 1000, 10000\}$ when $n_{\text{tar}} = 500$.

Fig. 4 shows the F1 scores and AUC of OCSTM with respect to parameter $\lambda'$ and the scores of the other compared methods. Note that scores of the compared methods are the best scores by varying their parameters. We see that the performance of OCSTM does not largely depend on the parameter $\lambda'$. Although when $\lambda' = 100$ the F1 score of OCSTM is lowest, the performance is still higher than the other methods.
5 CONCLUSIONS

In this paper, we proposed a one-class transfer learning method named OCSTM, for personalized anomalous facial expression detection. Unlike other anomaly detection methods, the OCSTM learns a personalized model from the target and source samples by re-weighting the samples based on their proximity to the target samples. Therefore, re-weighted samples help the target model to avoid overfitting even if the sample size of the target samples is small, and the classifier handles the individual differences appropriately. Experiments conducted on UNBC-MSPEA database show that OCSTM outperforms original one-class SVM including the generic and single-task model, and the state-of-the-art ML method. Furthermore, since the selection of feature extraction methods highly influences the performance of one-class methods, we investigated suitable features for OCSTM in anomalous facial expression detection. The results show that DFL produces higher accuracies than LBPH and SIFTD because low-dimension features, i.e., DFL, contain less irrelevant information than high-dimension ones, i.e., LBPH and SIFTD.

ACKNOWLEDGEMENTS

This work was partially supported by JSPS KAKENHI Grant Number JP15K12100.

REFERENCES


