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 coveys 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

Fujita, H., Matsukawa, T. and Suzuki, E.
One-class Selective Transfer Machine for Personalized Anomalous Facial Expression Detection.
DOI: 10.5220/0006613502740283
In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018) - Volume 5: VISAPP*, pages 274-283
ISBN: 978-989-758-290-5

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, feat, 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 visually 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 $\mathbf{X}^{\mathrm{sc}} = \{\mathbf{x}_i^{\mathrm{sc}}\}_{i=1}^{n_{\mathrm{sc}}}$, and the target samples $\mathbf{X}^{\mathrm{tar}} = \{\mathbf{x}_i^{\mathrm{tar}}\}_{i=1}^{n_{\mathrm{tar}}}$, where $\mathbf{x}_i^{\mathrm{sc}}$, $\mathbf{x}_i^{\mathrm{tar}} \in \mathbb{R}^d$, 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 $f(\mathbf{x}^{\mathrm{tar}})$ 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 $f(\mathbf{x}^{\mathrm{tar}}) = \mathrm{sign}(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}^{\mathrm{tar}}) - \rho)$, where $\phi(\cdot)$ is the nonlinear feature mapping associated with a kernel function $k(\mathbf{x},\mathbf{y}) = \phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{y})$, and $\mathbf{w}$, ρ are parameters of a hyperplane. In most of the kernel functions such as Gaussian kernel, the mapped example $\phi(\mathbf{x})$ cannot be calculated explicitly (Amari and Wu, 1999) and thus we cannot obtain the hyperplane parameter $\mathbf{w}$ explicitly. Instead, we obtain the inner products between the hyperplane parameter $\mathbf{w}$ and the mapped samples $\phi(\mathbf{x})$ 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:

$$(\mathbf{w},\mathbf{s}) = \arg\min_{\mathbf{w},\mathbf{s}} R_{\mathbf{w}}(\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}},\mathbf{s}) + \lambda\Omega_{\mathbf{s}_{\mathrm{sc}}}(\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}}), \tag{1}$$

where $R_{\mathbf{w}}(\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}},\mathbf{s})$ is the empirical risk (details are given in Sec. 3.1.1) defined on the source and target samples $\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}}$ with each instance $\mathbf{x}^{\mathrm{sc}}$ and $\mathbf{x}^{\mathrm{tar}}$ weighted by $\mathbf{s}^{\mathrm{sc}} \in \mathbb{R}^{n_{\mathrm{sc}}}$ and $\mathbf{s}^{\mathrm{tar}} \in \mathbb{R}^{n_{\mathrm{tar}}}$, respectively. Each element s_i^{sc} and s_i^{tar} corresponds to a non-negative weight for the sample $\mathbf{x}_i^{\mathrm{sc}}$ and $\mathbf{x}_i^{\mathrm{tar}}$, respectively. We denote $\mathbf{s}$ as the vertical concatenation of $\mathbf{s}^{\mathrm{sc}}$ and $\mathbf{s}^{\mathrm{tar}}$ by $\mathbf{s} = (s_1^{\mathrm{sc}},...,s_{n_{\mathrm{sc}}}^{\mathrm{sc}},s_1^{\mathrm{tar}},...,s_{n_{\mathrm{tar}}}^{\mathrm{tar}})^{\mathrm{T}}$. The second term $\Omega_{\mathbf{s}^{\mathrm{sc}}}(\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}})$ measures the distribution discrepancy between the source and target distributions as a function of $\mathbf{s}^{\mathrm{sc}}$ (details are given in Sec. 3.1.2). The lower the value of $\Omega_{\mathbf{s}_{\mathrm{sc}}}(\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}})$, the more similar the source and target distributions are. A parameter λ (≥ 0) balances the empirical risk term and the distribution discrepancy term.

3.1.1 Empirical Risk

The first term in Eq. (1), $R_{\mathrm{w}}(\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}},\mathbf{s})$, 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 $\nu \in (0,1)$, 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:

$$R_{\mathbf{w}}(\mathbf{X}^{\mathrm{sc}},\mathbf{X}^{\mathrm{tar}},\mathbf{s}) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{\nu n_{\mathrm{all}}}\sum_{i=1}^{n_{\mathrm{all}}} s_i\xi_i - \rho,$$
$$\text{s.t. } \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) \geq \rho - \xi_i,\ \xi_i \geq 0, \tag{2}$$

where $n_{\mathrm{all}} = n_{\mathrm{sc}} + n_{\mathrm{tar}}$, $\{\mathbf{x}_i\}_{i=1}^{n_{\mathrm{all}}} = \mathbf{X}^{\mathrm{sc}} \cup \mathbf{X}^{\mathrm{tar}}$, and ξ_i is a slack variable for training sample $\mathbf{x}_i$. If ξ_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 ξ_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), $\Omega_{\mathbf{s}^{\mathrm{sc}}}(\mathbf{X}^{\mathrm{sc}}, \mathbf{X}^{\mathrm{tar}})$, 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 $\mathbf{s}^{\mathrm{sc}}$ that minimizes

$$\Omega_{\mathbf{s}^{\mathrm{sc}}}(\mathbf{X}^{\mathrm{sc}}, \mathbf{X}^{\mathrm{tar}}) = \left\| \frac{1}{n_{\mathrm{sc}}} \sum_{i=1}^{n_{\mathrm{sc}}} s_i^{\mathrm{sc}} \phi(\mathbf{x}_i^{\mathrm{sc}}) - \frac{1}{n_{\mathrm{tar}}} \sum_{j=1}^{n_{\mathrm{tar}}} \phi(\mathbf{x}_j^{\mathrm{tar}}) \right\|_{\mathcal{H}}^2, \quad (3)$$

where $\|\cdot\|_{\mathcal{H}}^2$ is L2-norm in RKHS. We introduce $\kappa_i := \frac{n_{\mathrm{sc}}}{n_{\mathrm{tar}}} \sum_{j=1}^{n_{\mathrm{tar}}} k(\mathbf{x}_i^{\mathrm{sc}}, \mathbf{x}_j^{\mathrm{tar}})$, $i = 1, ..., n_{\mathrm{sc}}$, which captures the proximity between the source and each target sample in $\mathcal{H}$, and two constraints $s_i^{\mathrm{sc}} \in [0, B]$, $\left| \frac{1}{n_{\mathrm{sc}}} \sum_{i=1}^{n_{\mathrm{sc}}} s_i^{\mathrm{sc}} - 1 \right| \leq \varepsilon$. 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 $\mathbf{x}_i^{\mathrm{sc}}$. For $B \to 1$, we obtain the unweighted solution. ε 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 $\mathbf{s}^{\mathrm{sc}}$ in Eq. (3) can be written as a quadratic programming (QP):

$$\min_{\mathbf{s}^{\mathrm{sc}}} \frac{1}{2} (\mathbf{s}^{\mathrm{sc}})^{\mathrm{T}} \mathbf{K}^{\mathrm{sc}} \mathbf{s}^{\mathrm{sc}} - \boldsymbol{\kappa}^{\mathrm{T}} \mathbf{s}^{\mathrm{sc}},$$
$$\text{s.t. } s_i^{\mathrm{sc}} \in [0, B], \left| \sum_{i=1}^{n_{\mathrm{sc}}} s_i^{\mathrm{sc}} - n_{\mathrm{sc}} \right| \leq n_{\mathrm{sc}} \varepsilon, \quad (4)$$

where $\mathrm{K}_{ij}^{\mathrm{sc}} := k(\mathbf{x}_i^{\mathrm{sc}}, \mathbf{x}_j^{\mathrm{sc}})$, $i, j = 1, ..., n_{\mathrm{sc}}$ and $\boldsymbol{\kappa} = (\kappa_1, ..., \kappa_{n_{\mathrm{sc}}})^{\mathrm{T}}$. A large value of κ_i indicates large importance of $\mathbf{x}_i^{\mathrm{sc}}$ and is likely to lead to large s_i^{sc}.

3.2 Optimization

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

Note that the way of optimizing $\mathbf{w}$ 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 ρ, OCSTM can not 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 $\mathbf{s}$ is fixed, the subproblem over $\mathbf{w}$ corresponds to the minimization of the empirical risk $R_{\mathbf{w}}(\mathbf{X}^{\mathrm{sc}}, \mathbf{X}^{\mathrm{tar}}, \mathbf{s})$ because $\Omega_{\mathbf{s}^{\mathrm{sc}}}(\mathbf{X}^{\mathrm{sc}}, \mathbf{X}^{\mathrm{tar}})$ does not depend on $\mathbf{w}$. Eq. (2) can be minimized with Lagrange multipliers $\alpha_i, \beta_i \geq 0$. The Lagrangian of Eq. (2) is given by:

$$L(\mathbf{w}, \boldsymbol{\xi}, \rho, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{\nu n_{\mathrm{all}}} \sum_{i=1}^{n_{\mathrm{all}}} s_i \xi_i - \rho - \sum_{i=1}^{n_{\mathrm{all}}} \alpha_i (\mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}_i) - \rho + \xi_i) - \sum_{i=1}^{n_{\mathrm{all}}} \beta_i \xi_i. \quad (5)$$

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

$$\mathbf{w} = \sum_{i=1}^{n_{\mathrm{all}}} \alpha_i \phi(\mathbf{x}_i),$$
$$\alpha_i = \frac{s_i}{\nu n_{\mathrm{all}}} - \beta_i,$$
$$\sum_{i=1}^{n_{\mathrm{all}}} \alpha_i = 1. \quad (6)$$

[1] In (Chu et al., 2013), the source and target domains are respectively called the training and target domains.

[2] The representer theorem proves that the optimal solution can be written as a linear combination of kernel functions evaluated at the training samples for the optimization problem on a loss function added a regularization term $\lambda\|\mathbf{w}\|^2$ (Chapelle, 2007).

The dual problem is obtained from Eqs. (5) and (6):

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{K}^{\mathrm{all}}\boldsymbol{\alpha},$$
$$\text{s.t. } 0 \leq \alpha_i \leq \frac{s_i}{\nu n_{\mathrm{all}}}, \sum_{i=1}^{n_{\mathrm{all}}} \alpha_i = 1, \tag{7}$$

where $\mathrm{K}_{ij}^{\mathrm{all}} := k(\mathbf{x}_i, \mathbf{x}_j),\ i, j = 1, ..., n_{\mathrm{all}}$. The subproblem is convex because $\mathbf{K}^{\mathrm{all}} \succeq 0$. For any sample $\mathbf{x}_i$ whose corresponding α_i and β_i are nonzero at the optimum, i.e., $0 < \alpha_i < s_i/(\nu n_{\mathrm{all}})$, two inequality constraints in Eq.(2) become equalities, i.e., $\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) - \rho + \xi_i = 0$ and $\xi_i = 0$. From any such $\mathbf{x}_i$, we can recover ρ by the following equation:

$$\rho = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) = \sum_{j=1}^{n_{\mathrm{all}}} \alpha_j k(\mathbf{x}_j, \mathbf{x}_i). \tag{8}$$

We also recover the slack variables $\boldsymbol{\xi} = (\xi_1^{\mathrm{sc}}, ..., \xi_{n_{\mathrm{sc}}}^{\mathrm{sc}}, \xi_1^{\mathrm{tar}}, ..., \xi_{n_{\mathrm{tar}}}^{\mathrm{tar}})^{\mathrm{T}}$ in Eq. (2) by considering two cases of α_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 $\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) \geq \rho$. If $0 < \alpha_i \leq s_i/(\nu n_{\mathrm{all}})$, the first inequality constraint in Eq. (2) becomes equality, i.e., $\xi_i = \rho - \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i)$. Finally, we can obtain $\boldsymbol{\xi}$ by the following equation, $\xi_i = \max(0, \rho - \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i)),\ i = 1, ..., n_{\mathrm{all}}$. Using $\mathbf{w}$ in Eq. (6), the classifier of OCSTM is obtained as follows:

$$f(\mathbf{x}) = \mathrm{sign}(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) - \rho)$$
$$= \mathrm{sign}\left(\sum_{i=1}^{n_{\mathrm{all}}} \alpha_i k(\mathbf{x}_i, \mathbf{x}) - \rho\right). \tag{9}$$

3.2.2 Optimization on $\mathbf{s}^{\mathrm{sc}}$

When $\mathbf{w}$ is fixed, we obtain the subproblem over $\mathbf{s}^{\mathrm{sc}}$ from Eqs. (2) and (4), which corresponds to the following QP:

$$\min_{\mathbf{s}^{\mathrm{sc}}} \frac{1}{2}(\mathbf{s}^{\mathrm{sc}})^{\mathrm{T}}\mathbf{K}^{\mathrm{sc}}\mathbf{s}^{\mathrm{sc}} + \left(\frac{1}{\lambda\nu n_{\mathrm{all}}}\boldsymbol{\xi}^{\mathrm{sc}} - \boldsymbol{\kappa}\right)^{\mathrm{T}}\mathbf{s}^{\mathrm{sc}},$$
$$\text{s.t. } 0 \leq s_i^{\mathrm{sc}} \leq B,\ n_{\mathrm{sc}}(1-\varepsilon) \leq \sum_{i=1}^{n_{\mathrm{sc}}} s_i^{\mathrm{sc}} \leq n_{\mathrm{sc}}(1+\varepsilon), \tag{10}$$

where $\boldsymbol{\xi}^{\mathrm{sc}} = (\xi_1^{\mathrm{sc}}, ..., \xi_{n_{\mathrm{sc}}}^{\mathrm{sc}})^{\mathrm{T}}$. The subproblem is convex because $\mathbf{K}^{\mathrm{sc}} \succeq 0$. 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 $\boldsymbol{\xi}^{\mathrm{sc}}$. The source samples which have large ξ_i^{sc} lead to small s_i^{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 $\mathbf{X}^{\mathrm{sc}}, \mathbf{X}^{\mathrm{tar}}$, parameters $\sigma, \nu, \lambda, B, \varepsilon$
Output: Hyperplane parameters $\mathbf{w}, \rho$ and instance-wise weights $\mathbf{s}$

```
Initialize ξ ← 0
while not converged do
    Obtain the instance-wise weights s^sc by solving the QP in Eq. (10)
    if first loop then
        s^tar ← max(s^sc)1
    end if
    Obtain the hyperplane parameters w, ρ by solving Eqs. (7) and (8)
end while
```

Algorithm 1 summarizes the OCSTM algorithm[3]. While the instance-wise weights $\mathbf{s}^{\mathrm{sc}}$ 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 $\mathbf{s}^{\mathrm{tar}}$. To give target samples large instance-wise weights, the elements of $\mathbf{s}^{\mathrm{tar}}$ are set to the maximum value of $\mathbf{s}^{\mathrm{sc}}$ after the first optimization of $\mathbf{s}^{\mathrm{sc}}$.

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 λ.

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 σ, a zero-value vector whose length is n_{all} by $\mathbf{0}$, a one-value vector whose length is n_{tar} by $\mathbf{1}$, and the function that finds the maximum element of an input vector by $\max(\cdot)$.

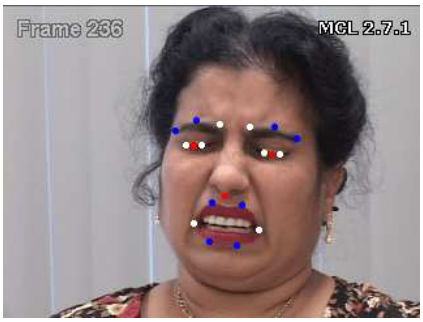


Figure 1: Sample in the UNBC-MSPEA database: A white or red point represents a landmark for landmark-based features and a white or blue point represents a landmark for SIFTD.

Table 1: Pairs of landmarks for computing DFL.

#	Pairs of landmarks
1	inner corners of right and left brows
2	right inner brow corner and nasal spine
3	left inner brow corner and nasal spine
4	right upper lid and lower lid
5	left upper lid and lower lid
6	right outer lid corner and right outer lip corner
7	left outer lid corner and left outer lip corner

25 patients performing a series of active and passive range-of-motion tests to their affected and unaffected limbs. All images in this dataset were annotated by Active Appearance Model (AMM) landmarks (Matthews and Baker, 2004) and the Prkachin and Solomon pain intensity (PSPI) metric (Prkachin and Solomon, 2008). A sample image and AAM landmarks are shown in Fig. 1. We used images of 10 subjects who exposed high intensity painful facial expressions (PSPI > 6). Low intensity painful facial expression images ($0 <$ PSPI ≤ 6) were not used. The number of used images were 19,429 including 383 painful facial expression images.

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 Transform Descriptors (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 200×200 pixel window. Then descriptors were computed within 36×36 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 128×128 pixel window. Then the resized face image was divided to 8×8 blocks and the LBP histograms were extracted from the blocks. We apply $uniform\ LBP^{u2}_{8,1}$ to each block, where u2 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 (AU 43). 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 ($n_{\mathrm{tar}} = 500$). Each row shows the result when the each subject was used for the test subject and the last row shows their average.

	$F1$ score				AUC			
subject	T-OCSVM	ST-OCSVM	ML-OCSVM	OCSTM	T-OCSVM	ST-OCSVM	ML-OCSVM	OCSTM
1	0.11	**0.27**	0.12	0.21	0.99	**1.00**	**1.00**	0.99
2	0.31	0	0.29	**0.54**	0.99	0.67	**1.00**	**1.00**
3	0.88	**0.93**	0.85	0.90	**1.00**	**1.00**	**1.00**	**1.00**
4	0.60	0	0.54	**0.61**	0.89	0.53	**0.90**	0.85
5	0	0	0	0	0.50	0.76	0.46	**0.77**
6	0.15	**0.23**	0.14	0.21	0.96	0.96	0.93	**0.97**
7	0.34	**0.67**	0.40	0.56	**1.00**	**1.00**	**1.00**	**1.00**
8	**0.75**	0.49	**0.75**	0.73	**0.95**	0.76	0.94	0.91
9	0.67	0	0.69	**0.82**	0.99	0.24	**1.00**	**1.00**
10	0	0	0	0	0.48	0.52	0.41	**0.72**
average	0.38	0.26	0.38	**0.46**	0.88	0.74	0.86	**0.92**

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 n_{tar} 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[4]. 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 $\nu = 0.0001$ 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 $\mathbf{s}$, $1 \leq \frac{1}{\nu n_{\mathrm{all}}} \sum_{i=1}^{n_{\mathrm{all}}} s_i$. To ensure this inequality through the second constraint in Eq. (10), we set $\varepsilon = 1 - \nu n_{\mathrm{all}} / n_{\mathrm{sc}}$.

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., $B = n_{\mathrm{sc}} / n_{\mathrm{cl}}$, where n_{cl} 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 $n_{\mathrm{cl}} = 0$, 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 n_{sc} to balance the slack variables $\boldsymbol{\xi}$ and $\boldsymbol{\kappa}$. In Eq. (10), the $\frac{1}{\lambda \nu n_{\mathrm{all}}} \boldsymbol{\xi}$ tends to be significantly smaller than $\boldsymbol{\kappa}$ due to the term $\frac{1}{n_{\mathrm{all}}}$ [5]. In addition, $\boldsymbol{\kappa}$ is weighted by n_{sc} by definition, i.e., $\kappa_i := \frac{n_{\mathrm{sc}}}{n_{\mathrm{tar}}} \sum_{j=1}^{n_{\mathrm{tar}}} k(\mathbf{x}_i^{\mathrm{sc}}, \mathbf{x}_j^{\mathrm{tar}})$. Since n_{sc} and n_{all} for the target person are different from those of the other persons, and n_{sc} is nearly equal to n_{all}, we set λ as $\lambda = \lambda' / n_{\mathrm{sc}}^2$. In Sec. 4.7, we investigated the dependency of OCSTM on the parameter λ'. Since the dependency is small, we set $\lambda' = 1{,}000$ in all experiments except in Sec. 4.7.

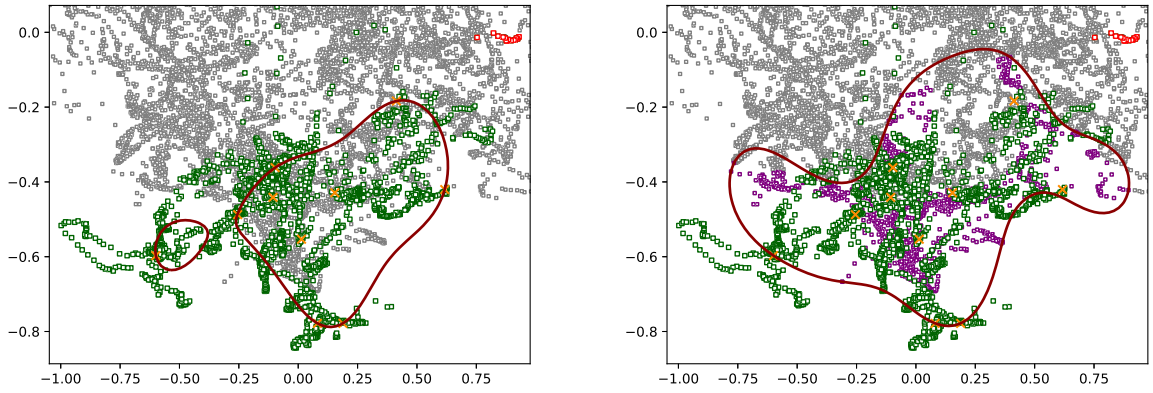
4.4 Comparison with Related Methods

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

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

[5] In the empirical risk of STM, the training loss is not weighted by n_{all}, i.e., $C \sum_{i=1}^{n_{\mathrm{tr}}} s_i L_p(y_i, \mathbf{w}^{\mathrm{T}} \mathbf{x}_i)$, where n_{tr} means the number of the source samples in this paper, $L_p(y, \cdot)$ is a loss function for each sample whose class label is y, and C is a constant parameter.



(a) T-OCSVM on target samples

(b) OCSTM on target and source samples

Figure 2: 2D PCA projections of the samples and the hyperplanes for T-OCSVM and OCSTM when $n_{\text{tar}} = 10$. 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.

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

subject	normal-anomalous	normal-normal
1	0.140	**0.331**
2	0.019	**0.358**
3	0.007	**0.345**
4	0.256	**0.372**
5	**0.411**	0.333
6	0.012	**0.316**
7	0.003	**0.361**
8	0.169	**0.321**
9	0.178	**0.406**
10	**0.418**	0.362

all the other methods on average. The higher scores of OCSTM compared with T-OCSVM are not surprising because T-OCSVM was learnt from only limited training samples and thus suffered from overfitting. As discussed in Sec. 1, ST-OCSVM cannot handle the individual differences appropriately, resulting in low $F1$ scores for several subjects. The scores of ML-OCSVM are close to those of T-OCSVM. This is because that ML-OCSVM combines the ST-OCSVM and T-OCSVM models and a higher combination weight for T-OCSVM was selected. In contrast to ML-OCSVM, which can only produce intermediate classifiers of two models, OCSTM fits the target distribution better since OCSTM selects source samples which are close to the target samples.

Table 3 shows the average similarities between the normal and anomalous samples in the target domain, and the similarity is given by the Gaussian kernel function. For subjects #5 and #10, the similarity between the normal and anomalous samples in the target domain is larger than that between the target normal samples. Therefore, the requirement (1) in Sec. 4.2 is violated, and $F1$ scores are zero (Table 2).

Fig. 2 shows 2D PCA projections of the samples and the hyperplanes for T-OCSVM and OCSTM. In Fig. 2, green and red squares respectively represent the normal and anomalous test samples in the target domain, and orange crosses and gray squares respectively represent the normal training samples in the target and source domains. Purple squares in Fig. 2 (b) represent the source samples that are given larger instance-wise weights than the mean. Since we used non-linear feature mapping, a hyperplane is a closed surface in the example space. A red closed surface represents a hyperplane of each classifier which predicts a sample inside as normal. In Fig. 2 (a) many normal test samples are outside the closed surface, which signifies that the model of T-OCSVM overfits to a few target samples. Conversely, in Fig. 2 (b) more normal test samples are inside the closed surface than T-OCSVM in Fig. 2 (a), which signifies that OCSTM avoids overfitting unlike T-OCSVM. Since in Eq. (2), small weights $\mathbf{s}$ are given to the slack variables $\boldsymbol{\xi}$ of the source samples which are far from the target samples, OCSTM hardly considers such samples. Consequently, the optimization in Eq. (1) yields a hyperplane such that the target samples and the source samples which are close to the target samples are inside the closed surface.

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 ($n_{\text{tar}} = 500$).

	T-OCSVM			OCSTM		
subject	LBPH	SIFTD	DFL	LBPH	SIFTD	DFL
1	0.03	0.03	**0.11**	0.07	0.05	**0.21**
2	0.11	0.13	**0.31**	0.20	0.18	**0.54**
3	0.64	0.69	**0.88**	0.72	0.75	**0.90**
4	0.45	0.52	**0.60**	0.59	0.59	**0.61**
5	0.06	**0.08**	0	0.09	**0.15**	0
6	0.06	0.06	**0.15**	0.08	0.08	**0.21**
7	0.11	0.14	**0.34**	0.18	0.18	**0.56**
8	0.52	0.59	**0.75**	0.60	0.62	**0.73**
9	0.35	0.41	**0.67**	0.44	0.43	**0.82**
10	**0.46**	0.45	0	**0.63**	0.52	0
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_{tar}. We conducted experiments using DFL by varying n_{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_{tar} decreases. OCSTM outperforms the other methods for each n_{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.

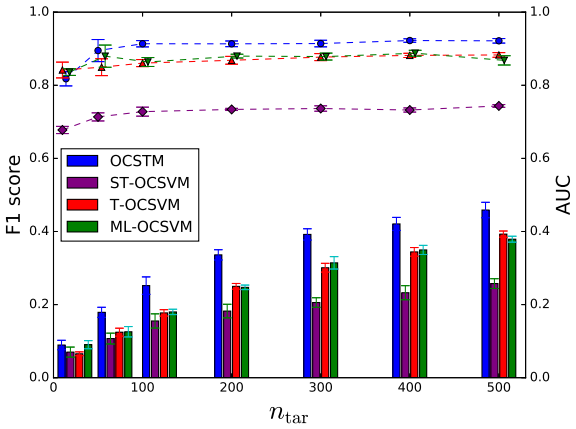


Figure 3: Performance of OCSTM with respect to the number n_{tar} of target samples. A line graph and a bar graph respectively represent $F1$ scores and AUC for each method.

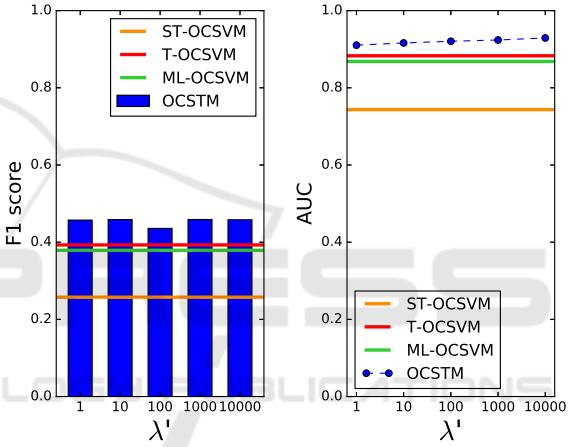


Figure 4: $F1$ scores and AUC of OCSTM with respect to the parameter λ' ($n_{\text{tar}} = 500$). For comparison, the scores of the other methods are shown as horizontal lines.

4.7 Dependency of OCSTM on the Parameter λ

Here we analyze how the performance of our method depends on the parameter $\lambda = \lambda'/n_{\text{sc}}^2$. 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 λ' 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 λ'. 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

Ahonen, T., Hadid, A., and Pietikäinen, M. (2006). Face Description with Local Binary Patterns: Application to Face Recognition. *IEEE Trans. on PAMI*, 28(12):2037–2041.

Amari, S. and Wu, S. (1999). Improving Support Vector Machine Classifiers by Modifying Kernel Functions. *Neural Networks*, 12(6):783–789.

Chapelle, O. (2007). Training a Support Vector Machine in the Primal. *Neural Computation*, 19(5):1155–1178.

Chen, J. and Liu, X. (2014). Transfer Learning with One-Class Data. *Pattern Recognition Letters*, 37:32–40.

Chen, J., Liu, X., Tu, P., and Aragones, A. (2013). Learning Person-Specific Models for Facial Expression and Action Unit Recognition. *Pattern Recognition Letters*, 34(15):1964–1970.

Chu, W.-S., Torre, F. D. L., and Cohn, J. F. (2013). Selective Transfer Machine for Personalized Facial Action Unit Detection. In *CVPR*, pages 3515–3522.

Ekman, P. and Friesen, W. V. (1978). *Facial Action Coding System*. Consulting Psychologists Press, Palo Alto, CA.

Gorski, J., Pfeuffer, F., and Klamroth, K. (2007). Biconvex Sets and Optimization with Biconvex Functions: a Survey and Extensions. *Mathematical Methods of Operations Research*, 66(3):373–407.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., and Schölkopf, B. (2009). *Covariate Shift by Kernel Mean Matching*, chapter 8, pages 131–160. MIT Press, Cambridge, MA.

He, X., Mourot, G., Maquin, D., Ragot, J., Beauseroy, P., Smolarz, A., and Grall-Maës, E. (2014). Multi-Task Learning with One-Class SVM. *Neurocomputing*, 133:416–426.

Lucey, P., Cohn, J. F., Prkachin, K. M., Solomon, P. E., and Matthews, I. (2011). Painful Data: The UNBC-McMaster Shoulder Pain Expression Archive Database. In *FG*, pages 57–64.

Matthews, I. and Baker, S. (2004). Active Appearance Models Revisited. *International Journal of Computer Vision*, 60(2):135–164.

Mohammadian, A., Aghaeinia, H., Towhidkhah, F., et al. (2016). Subject Adaptation Using Selective Style Transfer Mapping for Detection of Facial Action Units. *Expert Systems with Applications*, 56:282–290.

Prkachin, K. M. and Solomon, P. E. (2008). The Structure, Reliability and Validity of Pain Expression: Evidence from Patients with Shoulder Pain. *Pain*, 139(2):267–274.

Sangineto, E., Zen, G., Ricci, E., and Sebe, N. (2014). We are not All Equal: Personalizing Models for Facial Expression Analysis with Transductive Parameter Transfer. In *ACM Multimedia*, pages 357–366.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., and Williamson, R. C. (2001). Estimating the Support of a High-Dimensional Distribution. *Neural Computation*, 13(7):1443–1471.

Shan, C., Gong, S., and McOwan, P. W. (2009). Facial Expression Recognition Based on Local Binary Patterns: A Comprehensive Study. *Image and Vision Computing*, 27(6):803–816.

Zen, G., Porzi, L., Sangineto, E., Ricci, E., and Sebe, N. (2016). Learning Personalized Models for Facial Expression Analysis and Gesture Recognition. *IEEE Trans. on Multimedia*, 18(4):775–788.

Zeng, J., Chu, W.-S., Torre, F. D. L., Cohn, J. F., and Xiong, Z. (2015). Confidence Preserving Machine for Facial Action Unit Detection. In *ICCV*, pages 3622–3630.

Zeng, Z., Fu, Y., Roisman, G. I., Wen, Z., Hu, Y., and Huang, T. S. (2006). One-Class Classification for Spontaneous Facial Expression Analysis. In *FG*, pages 281–286.

Zeng, Z., Pantic, M., Roisman, G. I., and Huang, T. S. (2009). A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions. *IEEE Trans. on PAMI*, 31(1):39–58.