Gaussian Process Domain Experts for Modeling of Facial Affect

Stefanos Eleftheriadis, Ognjen (Oggi) Rudovic, *Member, IEEE,*
Marc Peter Deisenroth and Maja Pantic, *Fellow, IEEE*

***Abstract*—Most of existing models for facial behavior analysis rely on generic classifiers, which fail to generalize well to previously unseen data. This is because of inherent differences in source (training) and target (test) data, mainly caused by variation in subjects' facial morphology, camera views, etc. All of these account for different contexts in which target and source data are recorded, and thus, may adversely affect the performance of the models learned solely from source data. In this paper, we exploit the notion of domain adaptation and propose a data efficient approach to adapt already learned classifiers to new unseen contexts. Specifically, we build upon the probabilistic framework of Gaussian processes (GPs), and introduce domain-specific GP experts (*e.g.*, for each subject). The model adaptation is facilitated in a probabilistic fashion, by conditioning the target expert on the predictions from multiple source experts. We further exploit the predictive variance of each expert to define an optimal weighting during inference. We evaluate the proposed model on three publicly available datasets for multi-class (MultiPIE) and multi-label (DISFA, FERA2015) facial expression analysis by performing adaptation of two contextual factors: 'where' (view) and 'who' (subject). In our experiments, the proposed approach consistently outperforms (i) both source and target classifiers, while using a small number of target examples during the adaptation, and (ii) related state-of-the-art approaches for supervised domain adaptation.**

***Index Terms*—domain adaptation, Gaussian processes, multiple AU detection, multi-view facial expression recognition.**

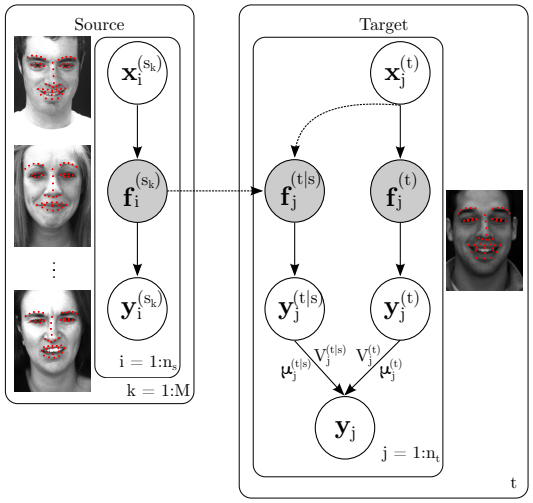


Fig. 1. The proposed GPDE model. The learning consists of training the multiple source $(s_k, k = 1, \cdots, M)$ and the target (t) GP experts (in this case, each subject is treated as an expert), using the available labeled training data pairs $(\boldsymbol{x}, \boldsymbol{y})$ – the input features (*e.g.*, facial landmarks) and output labels (*e.g.*, AU activations), respectively. Adaptation (dashed lines) for the target data is performed via conditioning the latent functions, $\boldsymbol{f}$, of the target GP on the source experts $(t|s)$. During inference, we fuse the predictions from the experts $(\boldsymbol{\mu}^{\{t,(t|s)\}})$ by means of their predictive variance $(V^{\{t,(t|s)\}})$, with the role of a confidence measure.

I. Introduction

THE human face is believed to be the most powerful channel for non-verbally conveying behavioral traits, such as personality, intentions and affect [1], [2]. Throughout the ages, people have learned to communicate the behavioral traits to their environment via their facial expressions. Facial expressions can be described at different levels [3]: The more prevalent approaches focus on identifying either the exact facial affect (emotions) or the activations of facial muscles, named action units (AUs). According to [4] these orthogonal approaches are just different measurements for facial expressions. A comprehensive system that can be used to unify the different measurements is the facial action coding system (FACS) [5]. FACS defines 30+ unique AUs and several categories of head/eye movements, which can be used to describe every possible facial expression.

Due to its practical importance in medicine, marketing and entertainment, automated analysis of facial expressions has received significant attention over the last two decades [6]. Despite rapid advances in computer vision and machine learning, the majority of the models proposed so far for facial expression analysis rely on *generic* classifiers. With the term 'generic' we refer to simple classifiers that are trained on all available data, which is assumed to encode all possible variations of the population. Hence, the performance of these classifiers is expected to degrade when applied to previously unseen data [7]. Such a scenario is the case when we try to infer the facial expression of a new subject whose level of expressiveness deviates substantially from the ones of the training subjects.

Besides the subject identity, there are also other sources of variation that can significantly affect the performance of generic classifiers. These sources can well be grouped accord-

S. Eleftheriadis was with the Department of Computing, Imperial College London, UK. He is now with PROWLER.io, Cambridge, UK. E-mail: stefanos@prowler.io

O. Rudovic is with the MIT Media Lab, USA. E-mail: orudovic@mit.edu

M. P. Deisenroth is with the Department of Computing, Imperial College London, UK and with PROWLER.io, Cambridge, UK. E-mail: m.deisenroth@imperial.ac.uk.

M. Pantic is with the Department of Computing, Imperial College London, UK and with the Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, The Netherlands. E-mail: m.pantic@imperial.ac.uk

Manuscript received xxxx; revised xxx.

ing to the W5+ context design [8], which describes the target behavior in terms of the context questions 'who', 'where', 'how', 'what', 'when' and 'why'. Ideally, an appropriate model for facial expression analysis should take into account all the above contextual factors during training. However, due to the lack of appropriate data, such an approach is not feasible. Thus, the majority of the work has focused only on building personalized classifiers for answering the question 'who' [8], [9], [10], [11], or on combining illumination invariant features with multi-view learning techniques for addressing the question 'where' (variations in head-pose and illumination) [12], [13], [14], [15], [16], [17]. Although these approaches showed improvement over generic classifiers, there remain a number of challenges to address. In particular, multi-view learning requires a large number of images in various poses, which are typically not readily available. Furthermore, for building personalized classifiers, access to an adequate collection of images of the target person is essential. Consequently, existing approaches perform re-weighting of previously learned classifiers to fit the target data (*e.g.*, [10]), or training of new models using additional target data. However, they are both sub-optimal since they require re-training of the original models.

A better solution would be to develop mechanisms that can adapt the learned models to the context of the examined situation. In this article, we propose a first step in this direction. In particular, we present an approach that can be used to adapt the context questions *where* (view) and *who* (subject), for facial expression recognition (FER) and AU detection, respectively. More specifically, we explore the problem of *domain adaptation*, where the distribution of the (facial) features varies across domains (*i.e.*, contexts such as the view or subject), while the output labels (*i.e.*, the emotion expression or the AU activations) remain the same. In the case of the context question 'where', this boils down to adapting the frontal classifier to a non-frontal view using only a small number of expressive images from the target view. Similarly, in the case of the subject adaptation ('who'), the model adaptation is performed by using as few annotated images of target subject as needed to gain in the prediction performance (*e.g.*, AU detection). Thus, our aim is to find a data-efficient approach to adapt previously trained generic models for facial behavior analysis, and overcome the burden of computation-wise costly model relearning.

The proposed model is a generalization of Gaussian processes (GPs) [18], and the product of expert models [19], [20], to the domain adaptation scenario.[1] More specifically, instead of adjusting the classifier parameters between the domains, as in [10], [21], [22], [23], [11], we propose the use of domain-specific GP experts that model the domain specific attributes. The modeling power of GPs allows us to model the desired attributes in the target domain, in a data-efficient manner. This is crucial for the training of the target expert since the available annotated data are usually scarce. Moreover, instead of minimizing the error between the distributions of the original source and target domain data, as in [10], [23], we use Bayesian domain adaptation [24] and facilitate the adaptation of the classifier by conditioning the target expert on the predictions from multiple source experts. The final prediction for the adapted classifier is obtained as a weighted combination of the predictions from the individual experts. The weighting is facilitated by measuring the confidence of each classifier. Contrary to [25] that represents the confidence heuristically as the agreement between a positive and a negative classifier, in our probabilistic formulation during the adaptation we exploit the variance in the GP predictions when combining the source and target domains [26]. This results in a *confident* classifier that minimizes the risk of potential negative transfer (*i.e.*, the adapted model performing worse than the model trained using the adaptation data only). Finally, in contrast to transductive adaptation approaches (*e.g.*, [10]) that need to be retrained completely, adaptation of our model is efficient and requires no retraining of the source model. An outline of the proposed model is depicted in Fig. 1. The contributions of this work can be summarized as follows:

- To the best of our knowledge, this is the first work in the field of facial behavior modeling that can simultaneously perform adaption to multiple outputs (*i.e.*, AUs). In our experiments, the proposed approach can effectively perform adaptation of 12 AUs, while existing models in the field attempt only adaptation for each output independently.
- Our proposed model exploits the variance in the predicted expression in order to utilize a measure of confidence for weighting the importance of each expert. This is in contrast to majority of the models that are purely discriminative and, thus, do not provide a probabilistic measure of 'reliability' for their predictions.
- Our approach is data efficient since it can perform the adaptation using only a small number of target labeled data. Through extensive experiments, we show empirically that it can generalize better than generic classifiers learned from the available source and/or target (training) data only, by using as few as 50 target samples for the adaptation.
- Our experiments demonstrate that the prediction mechanism based on the weighted combination of the source and target experts acts as a guard against negative transfer, allowing the model to explore the full capacity of the appropriate domain.

In our previous work [27], each output was constrained to have the same variance in its predictions. In this article, we relax this assumption by allowing each output to have a different confidence in the output. In case of AUs, this is a more realistic scenario since the proposed classifier may be more confident in predicting some AUs than the others. Hence, the weighting of the GP experts is decoupled across the multiple outputs, which results in more robust predictions when dealing with imbalanced datasets. Additional within- and cross-dataset experimental evaluations demonstrate the cases where the proposed re-weighted predictions are advantageous over [27].

[1] We use the non-parametric probabilistic framework of GPs as a basis for our model because it is particularly suited for learning highly non-linear mapping functions that can generalize from a small amount of training data.

II. Related Work

In this section, we first review the related work in facial behavior analysis. Then we discuss relevant machine learning approaches for domain adaptation.

A. Domain Adaptation in Facial Behavior Analysis

An important issue for the facial behavior analysis, and, in particular, the analysis of AUs, remains the poor generalizability to previously unseen data / contexts. Most works have attempted to address this issue by normalizing the data based on person-specific attributes (*e.g.*, removing the global neutral expression from an expressive image), as in [28]. However, recent advances in the field focus on employing standard domain adaptation techniques in order to build personalized classifiers for the test subjects. A widely used algorithm for adaptation is the kernel mean matching (KMM) [29], which directly infers resampling weights by matching training and test distributions. The authors in [10] employed the KMM to learn person-specific AU detectors. This is attained by modifying the SVM cost function to account for the mismatch in the distribution between source and target domain, while also adjusting the SVM's hyper-plane to the target test data. Although effective, this transductive learning approach is inefficient since for each target subject a new classifier has to be relearned during inference. Likewise, the authors in [23] proposed a supervised extension to the KMM. Specifically, they used the labeled examples from both domains in order to align the source and target distributions in a class-to-class manner. The reweighted source data along with the target data, form the input features that are used to train several classifiers.

Apart from KMM, adaptation can be also attained by combining the knowledge from multiple classifiers or by sharing the parameter space between source and target classifiers. In [22], a two-step learning approach is proposed for person-specific pain recognition and AU detection. First, data of each subject are regarded as different source domains, and are used to train weak Adaboost classifiers. Then, the weak classifiers are weighted based on their classification performance on the available target data. A second boosting is applied on the best performing source classifiers to derive the final set of weak classifiers for the target data. In [11], [21], the Adaboost classifiers are replaced with the linear SVMs. First, independent AU classifiers are trained from the source domain data. Then, the support vector regression is employed to associate the input features with the classifiers' parameters. Finally, the unlabeled target domain data are fed into the learned regressors, in order to obtain the target-specific classifiers parameters.

Recently, an attempt closer to our proposed method has been presented in [25]. The authors suggested to train target-specific classifiers by exploiting the confidence in the predictions from the source classifiers. In their approach, the confidence is represented by the agreement in the predictions between a pair of SVM classifiers, trained to distinguish the positive and negative samples in the source data. The confident classifiers are then employed to obtain 'virtual' labels for a portion of the target data, which can be used to train a target-specific detector.

Note that, apart from [22], all the works mentioned above operate in the unsupervised setting. While this requires less effort in terms of obtaining the labels for the target subsample, its underlying assumption is that target data can be well represented as a weighted combination of the source data. However, in order for this to work effectively, it is usually required to have access to lots of data from the target domain. Even when this is the case, in real world this assumption can easily be violated (*e.g.*, due to variations in subject's expressiveness, illuminations, etc.), resulting in poor performance of the adapted classifier.

In this work, we adopt a supervised approach that needs only a small amount of annotated data from target domain to perform the adaptation. This, in turn, allows us to define both target and source experts, assuring that the performance of the resulting classifier is not constrained by the distribution of the source data, as in unsupervised adaptation approaches. Contrary to transductive learning approaches such as [10], our approach requires adaptation of the target expert solely, without the need to relearn the source experts, resulting in an efficient adaptation process. Moreover, compared to our approach, only [25] provides a measure of confidence in the predicted labels. Yet, even in [25] the confidence is obtained in a heuristic manner and is not directly related to the prediction of the classifier. On the contrary, we model the confidence in a principled manner by means of predicted variance. Finally, note that the proposed approach and the methods mentioned above differ from those recently proposed for transfer learning, *e.g.*, [30]. The goal of the latter is to adapt a classifier learned for instance for one AU to another, which is different from the adaptation task addressed here and is out of the scope of this work.

B. Domain Adaptation

Domain adaptation is a well studied problem in machine learning (for an extensive survey, see [31]). In general, the adaptation problem stems from the change in the distributions of the input features and/or output labels between the two domains. The goal of domain adaptation is to match the differing distributions in order to learn a machinery that works sufficiently well on the test (target) data. Recent work has shown that the study of the causal relations between the data could be further useful on understanding how the distributions change across domains [32], [33]. The adaptation can be performed either in an *unsupervised* or a *(semi-)supervised* setting, based on the availability of labeled target domain data. The (semi-)supervised setting is more appropriate to our target task, since the available labels can be used to enhance the classification performance. One of the first attempts toward this directions has been presented in [34]. The authors proposed to replicate the input features to produce shared and domain-specific features, which are then fed into a generic classifier. Although straightforward, this approach has been proven effective for the adaptation task. [35] learns a transformation that maximizes similarity between data in the source and target domains by enforcing data pairs with the same labels to have high similarity, and pairs with different labels to be dissimilar. Then, a k-NN classifier is used to perform classification of target data. [36] is

an extension of this approach to multiple source domains. The input data are assumed to be generated from category-specific local domain mixtures, the mixing weights of which determine the underlying domain of the data, classified using an SVM classifier. Similarly, [37] learns a linear asymmetric transformation to maximally align target features to the source domain. This is attained by introducing max-margin constraints that allow the learning of the transformation matrix and SVM classifier jointly. [38] extends the work in [37] by introducing additional constraints to the max-margin formulation. More specifically, unlabeled data from the target domain are used to enforce the classifier to produce similar predictions for similar target-source data. While these methods attempt to directly align the target to source features, several works attempted this through a shared manifold. For instance, [39] learns a non-linear transformation from both source and target data to a shared latent space, along with the target classifier. Likewise, [40] finds a low-dimensional subspace, which preserves the structure across the domains. The subspace is facilitated by projections that are learned jointly with the linear classifier. The structure preservation constraints are used to ensure that similar data across domains are close in the subspace.

All of the methods mentioned above tackle the adaptation problem in a deterministic fashion. Thus, they do not provide a measure of confidence in the target predictions. By contrast, our approach is fully probabilistic and non-parametric due to the use of GPs, and is more related to recent advances in the literature [41], [24], [42] that perform the domain adaptation in a Bayesian fashion. Specifically, in [41] a discriminative framework is proposed to couple data from different domains in a shared subspace. Task-specific projections are learned simultaneously with the classifiers in order to couple all the task from the multiple domains in the obtained subspace. In [24], the predictive distribution of a GP trained on the source data is used as a prior for the joint distribution of the source and target domains. The information from the source domain can be analytically propagated to the inference of the target data by simply following the conditional properties of the GPs. Similarly, in [42] the authors proposed a two-layer GP that jointly learns separate discriminative functions from the source and target features to the labels. The intermediate layer facilitates the adaptation step and a variational approximation is employed to integrate out this layer, and propagate the information from the source to the target classifier.

Compared to the aforementioned work, our approach has the following key differences: in [41], the authors learn the classifier on a subspace shared among the data from source and target domains. This can be problematic in cases where access to target domain data is confined, since it bias the manifold toward explaining the variations from the source domain. In contrast to [24], our proposed approach defines a target specific expert, which is then combined with the source domain experts. The benefit of this is that the resulting classifier is not limited by the distribution of the source data. Also, in contrast to [42], the training of the experts is performed independently, and thus, we need not retrain the source classifier. Taken together, these differences bring significant improvements in estimation of the target tasks, as shown in our experiments.

III. Problem Formulation

We consider a supervised setting for domain adaptation, where we have access to a large collection of labeled *source* domain data, $\mathcal{S}$, and a smaller set of labeled *target* domain data, $\mathcal{T}$. Let $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ be the input (features) and output (labels) spaces, respectively. Hence, $\boldsymbol{X}^{(s)} = \{\boldsymbol{x}_{n_s}^{(s)}\}_{n_s=1}^{N_s}$ and $\boldsymbol{X}^{(t)} = \{\boldsymbol{x}_{n_t}^{(t)}\}_{n_t=1}^{N_t}$, with $\boldsymbol{x}_{n_s}^{(s)}, \boldsymbol{x}_{n_t}^{(t)} \in \mathbb{R}^D$, and $N_t \ll N_s$. In our case, the different domains can be different views or subjects. On the other hand, $\boldsymbol{Y}^{(s)} = \{\boldsymbol{y}_{n_s}^{(s)}\}_{n_s=1}^{N_s}$ and $\boldsymbol{Y}^{(t)} = \{\boldsymbol{y}_{n_t}^{(t)}\}_{n_t=1}^{N_t}$ correspond to same labels for both source and target domains. Each vector $\boldsymbol{y}_n^{\{s,t\}}$ contains the binary class labels of C classes. In order to avoid the burden of learning approximate solutions with GP classification, we formulate the predictions as a regression problem where:

$$\boldsymbol{y}_{n_v}^{(v)} = f^{(v)}(\boldsymbol{x}_{n_v}^{(v)}) + \epsilon^{(v)}, \tag{1}$$

where $\epsilon^{(v)} \sim \mathcal{N}(0, \sigma_v^2)$ is i.i.d. additive Gaussian noise, and the index $v \in \{s, t\}$ denotes the dependence on each domain. The objective is to infer the latent functions $f^{(v)}$, given the training dataset $\mathcal{D}^{(v)} = \{\boldsymbol{X}^{(v)}, \boldsymbol{Y}^{(v)}\}$. By following the framework of GPs [18], we place a prior on the functions $f^{(v)}$, so that the function values $\boldsymbol{f}_{n_v}^{(v)} = f^{(v)}(\boldsymbol{x}_{n_v}^{(v)})$ follow a Gaussian distribution $p(\boldsymbol{F}^{(v)}|\boldsymbol{X}^{(v)}) = \mathcal{N}(\boldsymbol{F}^{(v)}|\boldsymbol{0}, \boldsymbol{K}^{(v)})$. Here, $\boldsymbol{F}^{(v)} = \{\boldsymbol{f}_{n_v}^{(v)}\}_{n_v=1}^{N_v}$, and $\boldsymbol{K}^{(v)} = k^{(v)}(\boldsymbol{X}^{(v)}, \boldsymbol{X}^{(v)})$ is the kernel covariance function, which is assumed to be shared among the label dimensions. In this work, we use the radial basis function (RBF) kernel

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma_f^2 \exp\Big(-\frac{1}{2\ell^2}\|\boldsymbol{x} - \boldsymbol{x}'\|^2\Big), \tag{2}$$

where $\{\ell, \sigma_f\}$ are the kernel hyper-parameters. The regression mapping can be fully defined by the set of hyper-parameters $\boldsymbol{\theta} = \{\ell, \sigma_f, \sigma_v\}$. Training of the GP consists of finding the hyper-parameters that maximize the log-marginal likelihood

$$\log p(\boldsymbol{Y}^{(v)}|\boldsymbol{X}^{(v)}, \boldsymbol{\theta}^{(v)}) = -\frac{1}{2}\mathrm{tr}\Big[(\boldsymbol{K}^{(v)} + \sigma_v^2\boldsymbol{I})^{-1}\boldsymbol{Y}^{(v)}\boldsymbol{Y}^{(v)T}\Big] - \frac{C}{2}\log|\boldsymbol{K}^{(v)} + \sigma_v^2\boldsymbol{I}| + \text{const.} \tag{3}$$

Given a test input $\boldsymbol{x}_*^{(v)}$ the predicted function evaluation $\boldsymbol{f}_*^{(v)}$ is given from the GP predictive distribution by conditioning on the training data $\mathcal{D}^{(v)}$ as $p(\boldsymbol{f}_*^{(v)}|\boldsymbol{x}_*^{(v)}, \mathcal{D}^{(v)}) = \mathcal{N}(\mu^{(v)}(\boldsymbol{x}_*^{(v)}), V^{(v)}(\boldsymbol{x}_*^{(v)}))$ with

$$\mu^{(v)}(\boldsymbol{x}_*^{(v)}) = \boldsymbol{k}_*^{(v)T}(\boldsymbol{K}^{(v)} + \sigma_v^2\boldsymbol{I})^{-1}\boldsymbol{Y}^{(v)} \tag{4}$$

$$V^{(v)}(\boldsymbol{x}_*^{(v)}) = k_{**}^{(v)} - \boldsymbol{k}_*^{(v)T}(\boldsymbol{K}^{(v)} + \sigma_v^2\boldsymbol{I})^{-1}\boldsymbol{k}_*^{(v)}, \tag{5}$$

where $\boldsymbol{k}_*^{(v)} = k^{(v)}(\boldsymbol{X}^{(v)}, \boldsymbol{x}_*^{(v)})$ and $k_{**}^{(v)} = k^{(v)}(\boldsymbol{x}_*^{(v)}, \boldsymbol{x}_*^{(v)})$. For convenience we denote $\boldsymbol{\mu}_*^{(v)} = \mu^{(v)}(\boldsymbol{x}_*^{(v)})$ and $V_{**}^{(v)} = V^{(v)}(\boldsymbol{x}_*^{(v)})$. Under this general formulation, we have the choice to learn either (i) independent functions $f^{(v)}$ or (ii) a universal function f that couples the data from the two domains. However, neither option allows us to explore the idea of domain adaptation: In the former we learn domain-specific models, while in the latter we simplify the problem by concatenating the data from the two domains. An alternative would be to merge the two approaches in order to achieve a better generalization, while also being able to model domain

specific attributes. Such a combined approach would allow us to obtain more robust predictions.

IV. Domain Conditioned GPs

In the following, we introduce the notion of domain adaptation in the framework of GPs. Then, we present a novel methodology to merge the above mentioned learning scenarios, in order to obtain a universal classifier with good generalization abilities and capable of modeling domain specific attributes for the target tasks.

A. GP Adaptation

A straightforward approach to obtain a model capable of performing inference on data from both domains is to assume the existence of a universal latent function with a single set of hyper-parameters $\boldsymbol{\theta}$. Thus, the authors in [24] proposed a simple, yet effective, three-step approach for GP adaptation (GPA):

1) Train a GP on the source data with marginal likelihood $p(\boldsymbol{Y}^{(s)}|\boldsymbol{X}^{(s)}, \boldsymbol{\theta})$ to learn the hyper-parameters $\boldsymbol{\theta}$. The posterior distribution is then given by Eqs. (4–5).
2) Use the obtained posterior distribution of the source data, as a prior for the GP of the target data $p(\boldsymbol{Y}^{(t)}|\boldsymbol{X}^{(t)}, \mathcal{D}^{(s)}, \boldsymbol{\theta})$.
3) Correct the posterior distribution to account for the target data $\mathcal{D}^{(t)}$ as well.

Now the conditional prior of the target data (given the source data) in the second step is given by applying Eqs. (4–5) on $\boldsymbol{X}^{(t)}$

$$\boldsymbol{\mu}^{(t|s)} = {\boldsymbol{K}_{st}^{(s)}}^T (\boldsymbol{K}^{(s)} + \sigma_s^2 \boldsymbol{I})^{-1} \boldsymbol{Y}^{(s)} \tag{6}$$

$$\boldsymbol{V}^{(t|s)} = \boldsymbol{K}_{tt}^{(s)} - {\boldsymbol{K}_{st}^{(s)}}^T (\boldsymbol{K}^{(s)} + \sigma_s^2 \boldsymbol{I})^{-1} \boldsymbol{K}_{st}^{(s)}, \tag{7}$$

where $\boldsymbol{K}_{tt}^{(s)} = k^{(s)}(\boldsymbol{X}^{(t)}, \boldsymbol{X}^{(t)}), \boldsymbol{K}_{st}^{(s)} = k^{(s)}(\boldsymbol{X}^{(s)}, \boldsymbol{X}^{(t)})$, and the superscript $t|s$ denotes the conditioning order. Given the above prior and a test input $\boldsymbol{x}_*^{(t)}$, the correct form of the adapted posterior after observing the target domain data is:

$$\mu_{ad}^{(s)}(\boldsymbol{x}_*^{(t)}) = \boldsymbol{\mu}_*^{(s)} + {\boldsymbol{V}_*^{(t|s)}}^T (\boldsymbol{V}^{(t|s)} + \sigma_s^2 \boldsymbol{I})^{-1} (\boldsymbol{Y}^{(t)} - \boldsymbol{\mu}^{(t|s)}) \tag{8}$$

$$V_{ad}^{(s)}(\boldsymbol{x}_*^{(t)}) = V_{**}^{(s)} - {\boldsymbol{V}_*^{(t|s)}}^T (\boldsymbol{V}^{(t|s)} + \sigma_s^2 \boldsymbol{I})^{-1} \boldsymbol{V}_*^{(t|s)}, \tag{9}$$

with $\boldsymbol{V}_*^{(t|s)} = k^{(s)}(\boldsymbol{X}^{(t)}, \boldsymbol{x}_*^{(t)}) - k^{(s)}(\boldsymbol{X}^{(s)}, \boldsymbol{X}^{(t)})^T (\boldsymbol{K}^{(s)} + \sigma_s^2 \boldsymbol{I})^{-1} k^{(s)}(\boldsymbol{X}^{(s)}, \boldsymbol{x}_*^{(t)})$.

Eqs. (8–9) show that final prediction in the GPA is the combination of the original prediction based on the source data only, plus a correction term. The latter shifts the mean toward the distribution of the target data and improves the model's confidence by reducing the predictive variance. Note that we originally constrained the model to learn a single latent function f for both conditional distributions $p(\boldsymbol{Y}^{(v)}|\boldsymbol{X}^{(v)})$ to derive the posterior for the GPA. However, this constraint implies that the marginal distributions of the data $p(\boldsymbol{X}^{(v)})$ are similar. This assumption violates the general idea of domain adaptation, where by definition, the marginals may have significantly different attributes (*e.g.*, input features from different observation views). In such cases, GPA could perform worse than an independent GP trained solely on the target data $\mathcal{D}^{(t)}$. One possible way to address this issue is to retrain the $\log p(\boldsymbol{Y}^{(t)}|\boldsymbol{X}^{(t)}, \mathcal{D}^{(s)}, \boldsymbol{\theta})$ of the GPA w.r.t. $\boldsymbol{\theta}$ [24]. This option will compensate for the differences in the distributions by readjusting the hyper-parameters. However, it comes with the price of retraining of the model. Furthermore, it does not allow for modeling domain-specific attributes since the predictions are still determined mainly from the source distribution.

B. GP Domain Experts (GPDE)

In the proposed approach, we assume that each expert is a GP that operates only on a subset of data, *i.e.*, $\mathcal{D}^{(s)}, \mathcal{D}^{(t)}$. Hence, we can follow the methodology presented in Sec. III in order to train domain-specific GPs and learn different latent functions, *i.e.*, hyper-parameters $\boldsymbol{\theta}^{(v)}$. Within the current formulation we treat the source domain as a combination of multiple source datasets (*e.g.*, subject-specific datasets) $\mathcal{D}^{(s)} = \{\mathcal{D}^{(s_1)}, \ldots, \mathcal{D}^{(s_M)}\}$, where M is the total number of source domains (datasets).

Training. Given the above mentioned data split and assuming conditional independence of the labels from each domain given the corresponding input features, the marginal likelihood can be approximated by

$$p(\boldsymbol{Y}^{\{s,t\}}, |\boldsymbol{X}^{\{s,t\}}, \boldsymbol{\theta}^{\{s,t\}}) = p(\boldsymbol{Y}^{(t)}|\boldsymbol{X}^{(t)}, \boldsymbol{\theta}^{(t)}) \prod_{k=1}^{M} p_k(\boldsymbol{Y}^{(s_k)}|\boldsymbol{X}^{(s_k)}, \boldsymbol{\theta}^{(s)}). \tag{10}$$

We share the set of hyper-parameters $\boldsymbol{\theta}^{(s)}$ across all the source domains. The intuition behind this is that in each source domain we may observe a different conditional distribution $p(\boldsymbol{Y}^{(s_k)}|\boldsymbol{X}^{(s_k)})$, yet after exploiting all the available datasets we can model the overall conditional $p(\boldsymbol{Y}^{(s)}|\boldsymbol{X}^{(s)})$ with a single set of hyper-parameters $\boldsymbol{\theta}^{(s)}$. However, this does not guarantee that we are also able to explain the target conditional $p(\boldsymbol{Y}^{(t)}|\boldsymbol{X}^{(t)})$ with the same hyper-parameters. Recall that in our domain adaptation scenario the marginals of the labels are the same $p(\boldsymbol{Y}^{(t)}) = p(\boldsymbol{Y}^{(s)})$. However, both the marginal distribution of the features $p(\boldsymbol{X}^{(t)})$ and the conditional distribution of the labels $p(\boldsymbol{Y}^{(t)}|\boldsymbol{X}^{(t)})$ have changed in the target domain. Thus, we also search for $\boldsymbol{\theta}^{(t)}$ for modeling the domain-specific attributes. Similar to Sec. III learning of the hyper-parameters is performed by maximizing

$$\log p(\boldsymbol{Y}^{\{s,t\}}, |\boldsymbol{X}^{\{s,t\}}, \boldsymbol{\theta}^{\{s,t\}}) = \log p(\boldsymbol{Y}^{(t)}|\boldsymbol{X}^{(t)}, \boldsymbol{\theta}^{(t)}) + \sum_{k=1}^{M} \log p_k(\boldsymbol{Y}^{(s_k)}|\boldsymbol{X}^{(s_k)}, \boldsymbol{\theta}^{(s)}), \tag{11}$$

where each log-marginal is computed according to Eq. (3). The above factorization, apart from facilitating learning of the domain experts, allows for efficient GP training even with larger datasets, as shown in [19]. Note that the source experts can be learned independently from the target, which allows our model to generalize to unseen target domains without retraining.

Predictions. Once we have trained the GPDE, we need to combine the predictions from each expert to form an overall prediction. To achieve so, we build upon the approach in [20],

where we further readjust the predictions from the source experts using the conditional adaptation from GPA. Hence, the predictive distribution is given by

$$p(\boldsymbol{f}_*^{(t)}|\boldsymbol{x}_*^{(t)},\mathcal{D}) = \prod_{k=1}^{M} p_k^{\beta_{s_k}}(\boldsymbol{f}_*^{(t)}|\boldsymbol{x}_*^{(t)},\mathcal{D}^{(s_k)},\mathcal{D}^{(t)},\boldsymbol{\theta}^{(s)})\cdot p^{\beta_t}(\boldsymbol{f}_*^{(t)}|\boldsymbol{x}_*^{(t)},\mathcal{D}^{(t)},\boldsymbol{\theta}^{(t)}), \quad (12)$$

where β_{s_k}, β_t control the contribution of each expert. In this work we equally weight the experts and normalize them such that $\beta_t + \sum \beta_{s_k} = 1$, as suggested in [19]. The predictive mean and variance are then given by

$$\boldsymbol{\mu}_*^{\text{gpde}} = V_*^{\text{gpde}}\left[\beta_t {V_*^{(t)}}^{-1}\boldsymbol{\mu}_*^{(t)} + \textstyle\sum_k \beta_{s_k} {V_{ad}^{(s_k)}}^{-1}\boldsymbol{\mu}_{ad}^{(s_k)}\right] \quad (13)$$

$$V_*^{\text{gpde}} = \left[\beta_t {V_*^{(t)}}^{-1} + \textstyle\sum_k \beta_{s_k} {V_{ad}^{(s_k)}}^{-1}\right]^{-1}. \quad (14)$$

At this point the contribution of the GPDE becomes clear: Eq. (13) shows that the overall mean is the sum of the predictions from each expert, weighted by their precision (inverse variance). Hence, the solution of the GPDE will favor the predictions of more confident experts. On the other hand, if the quality of a domain expert is poor (noisy predictions with large variance), GPDE will weaken its contribution to the overall prediction.

C. Weighted GP Domain Experts for imbalanced outputs

In the analysis we conducted so far, we treated the multiple outputs as i.i.d. samples from a joint Gaussian distribution. Hence, we assumed a shared covariance matrix among the multiple output dimensions, which results in the same weighting/variance in Eqs. (13–14). This assumption becomes unrealistic in cases where we have to deal with imbalanced data in the output, *e.g.*, AUs with different occurrence patterns. Thus, it is important in each expert to account for a different variance per output. To address this, we follow the approach presented in [43], [44], and introduce a weighting matrix to the log-marginal likelihood of each expert in Eq. (11), so that

$$\log p(\boldsymbol{Y}^{(v)}|\boldsymbol{X}^{(v)},\boldsymbol{\theta}^{(v)}) = -\frac{1}{2}\text{tr}\left[(\boldsymbol{K}^{(v)}+\sigma_v^2\boldsymbol{I})^{-1}\boldsymbol{Y}^{(v)}\boldsymbol{\Lambda}^{(v)}{\boldsymbol{Y}^{(v)}}^T\right] - \frac{C}{2}\log|\boldsymbol{K}^{(v)}+\sigma_v^2\boldsymbol{I}| + \frac{N_v}{2}\log|\boldsymbol{\Lambda}^{(v)}| + \text{const}, \quad (15)$$

where $\boldsymbol{\Lambda}^{(v)} = \text{diag}(\lambda_1^{(v)},\cdots,\lambda_C^{(v)})$. This is equivalent to learning a GP with covariance function $k^{(v)}(\cdot,\cdot) = k^{(v)}(\cdot,\cdot)/\lambda_c^{(v)}$ for each output dimension c. The term $1/\lambda_c^{(v)}$ accounts for the different variances in the output dimensions and gives more flexibility to the model, since more representative input-output mappings can be learned.

Note, however, that the predicted variance of a probabilistic model depends highly on the training data. A GP domain expert can have access to data with zero activations for a certain output, while other outputs may frequently co-occur together. This suggests that there exists an intrinsic structure between the outputs, which we do not account for within the GPDE. To ameliorate this, we re-parameterize $\lambda_c^{(v)}$ as

$$\frac{1}{\lambda_c^{(v)}} = \frac{w_c^{(v)}}{\sum_c w_c^{(v)}}, \quad (16)$$

Algorithm 1 Domain adaptation with (w)GPDE

Inputs: $\mathcal{D}^{(s)} = \{\boldsymbol{X}^{(s)},\boldsymbol{Y}^{(s)}\}, \mathcal{D}^{(t)} = \{\boldsymbol{X}^{(t)},\boldsymbol{Y}^{(t)}\}$
Training:
Learn the hyper-parameters $\boldsymbol{\theta}^{\{s,t\}}$ by maximizing Eq. (11).
Adaptation:
Adapt the posterior from the source experts via Eq. (8–9).
Predictions of Experts:
Combine the prediction from each GP domain expert via Eq. (13–14) for GPDE or Eq. (17–18) for wGPDE.
Output: $\boldsymbol{y}_* = sign(\boldsymbol{\mu}_*^{gpde})$.

where $w_c^{(v)}$ is the new parameter to learn. As we can see from Eq. (16), the variance of each output is now proportional to the amount of the total variance. Such a re-parameterization correctly enforces the total variance of the GP to be distributed to the various outputs. It can be also regarded as a straightforward way to rectify the assumption of having i.i.d. outputs, since now frequently co-occurring outputs will be assigned similar weights, and, hence, a similar covariance function. We name this approach as *weighted* Gaussian process domain experts (wGPDE), to differentiate it from the single variance GPDE.

Re-weighted Predictions. By propagating the weighting matrix $\boldsymbol{\Lambda}$ to the predictive distribution of the proposed wGPDE, we can derive the re-weighted predictions for the c-th output

$$\boldsymbol{\mu}_{*_c}^{\text{gpde}} = V_{*_c}^{\text{gpde}}\left[\beta_t\lambda_c^{(t)}{V_*^{(t)}}^{-1}\boldsymbol{\mu}_{*_c}^{(t)} + \textstyle\sum_k \beta_{s_k}\lambda_c^{(s_k)}{V_{ad}^{(s_k)}}^{-1}\boldsymbol{\mu}_{ad_c}^{(s_k)}\right] \quad (17)$$

$$V_{*_c}^{\text{gpde}} = \left[\beta_t\lambda_c^{(t)}{V_*^{(t)}}^{-1} + \textstyle\sum_k \beta_{s_k}\lambda_c^{(s_k)}{V_{ad}^{(s_k)}}^{-1}\right]^{-1}. \quad (18)$$

By comparing Eqs. (13–14) to Eqs. (17–18) we see that the combined predictions from all the experts depend on the predicted variance of each output. This allows the re-weighted experts to be confident (higher contribution to the overall prediction) for certain outputs, while remaining 'silent' for outputs that have not seen. On the contrary, Eqs. (13–14) assign the same weight to all outputs, a fact that increases the bias in the predictions. Algorithm 1 summarizes the adaptation procedure of the proposed (w)GPDE.

V. Experiments

Datasets: We evaluate the proposed model on acted and spontaneous facial expressions from three publicly available datasets: MultiPIE [45], Denver Intensity of Spontaneous Facial Actions (DISFA) [46] and BP4D [47] (using the publicly available data subset from the FERA2015 [48] challenge). Specifically, MultiPIE contains images of 373 subjects depicting acted facial expressions of Neutral (NE), Disgust (DI), Surprise (SU), Smile (SM), Scream (SC) and Squint (SQ), captured at various pan angles. In our experiments, we used images from $0°$, $-15°$ and $-30°$. DISFA is widely used in the AU-related literature, due to the large amount of (subjects and AUs) annotated images. It contains video recordings of 27 subjects while watching YouTube videos. Each frame is coded in terms of the intensity of 12 AUs on a six-point ordinal scale. In our experiments, we treated each AU with intensity larger than zero as active. FERA2015 database includes videos of 41 participants. There are 21 subjects in the training and 20 subjects in the development

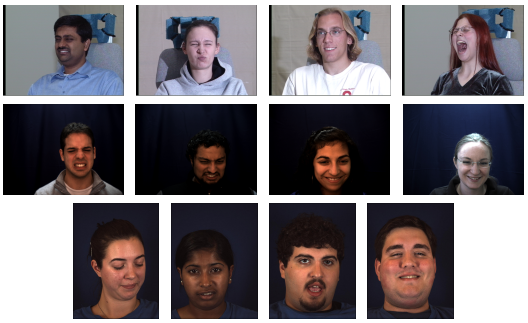


Fig. 2. Example images from MultiPIE (top), DISFA (middle) and FERA2015 (bottom) datasets.

partition. Each video is annotated in terms of occurrence of 11 AUs. Example images of the three datasets are given in Fig. 2.

Features: We use both a set of geometric features derived from the facial landmark locations, as well as appearance features. Specifically, DISFA and FERA2015 datasets come with frame-by-frame annotations of 66 and 49 facial landmarks, respectively, while a set of 66 annotated points for MultiPIE were obtained from [49]. After removing the contour landmarks from DISFA and MultiPIE annotations, we end up with the same set of 49 facial points for all three datasets. These were then registered to a reference face (average face per view for MultiPIE, and average face for DISFA and FERA2015) using an affine transformation. We then extract Local Binary Patterns (LBP) histograms [50] with 59 bins from patches centered around each registered point. Hence, we obtain 98D (geometric) and 2891D (appearance) feature vectors, commonly used in modeling of facial affect. For the high dimensional appearance features, in order to remove potential noise and artifacts, and also reduce the dimensionality, we applied PCA, retaining 95% of the energy, which resulted in approximately 200D appearance feature vectors.

Evaluation procedure. We evaluate (w)GPDE on both multi-class (FER on MultiPIE) and multi-label (multiple AU detection on DISFA and FERA2015) scenarios. We also assess the adaptation capacity of the model with a single (view adaptation) and multiple (subject adaptation) source domains. For the task of FER, images from $0°$, $-15°$ and $-30°$ served interchangeably as the source domain, while inference was performed via adaptation to the remaining views. For the AU detection task, the various subjects from the training data were used as multiple source domains, and adaptation was performed each time to the tested subject.

To evaluate the model's adaptation ability, we strictly follow a training protocol, where for each experiment we vary the cardinality of the training target data (we always use all the available source domain data). For MultiPIE, we first split the data in 5-folds (4 training, 1 testing and iterate over all folds) and then, we keep increasing the cardinality as: $N_t = 10, 30, 50, 100, 200, 300, 600, 1200$. For DISFA we follow a leave-one-subject-out approach (26 training source subjects and 1 target test subject at a time). For FERA2015 we followed the original partitioning suggested in [48] (20 training source subjects from the training partition, while each of the 20 subjects in the development partition served as an individual target domain). From the test subject's sequence in DISFA and FERA2015 the first 500 frames were used as target training data (with increasing cardinality $N_t = 10, 30, 50, 100, 200, 500$), while inference was performed on the rest frames of the sequence. This is in order to avoid the target model overfitting the temporally neighboring examples of the test subject. For the FER experiments, we employ the classification ratio (CR) as the evaluation measure, while for the AU detection we report the F1 score and the area under the ROC curve (AUC). Both F1 and AUC are widely used in the literature as they quantify different characteristics of the classifiers performance. Specifically, F1, defined as $F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$, is the harmonic mean between the precision and recall. It puts emphasis on the classification task, while being largely robust to imbalanced data (such as examples of different AUs). AUC quantifies the relation between true and false positives, showing the robustness of a classifier to the choice of its decision threshold.

Models compared. We compare the proposed approach with the two generic models GP_{source} and GP_{target}. The former is trained solely on the source data, while the latter on the target data used for the adaptation. We also compare to the modeled trained on the concatenation of the source and target training data, *i.e.*, GP_{s+t}. Additionally, we compare to the state-of-the-art models based on GPs for supervised domain adaptation, *i.e.*, the GPA [24] and the asymmetric transfer learning with deep GP (ATL-DGP) [42]. The GPA is an instance of the proposed GPDE, with only a source expert (no target) and predictions given by Eqs. (8)–(9). ATL-DGP employs an intermediate GP to combine the predictions of GP_{source} and GP_{target}. In Table I, we summarize the learning and inference complexity of all the GP-related methods. It is worth noting that GPA [24] and the proposed (w)GPDE can benefit from offline learning of the expensive source classifier, GP_{source}. GPA can perform directly the adaptation during inference. Hence, it is the most efficient method of all. However, the fact that it does not update the kernel's hyper-parameters after observing the target training data is the reason why it is expected most of the time to perform worse than the concatenated model, *i.e.*, GP_{s+t}. Adaptation in the proposed (w)GPDE depends only on the amount of available target training data N_t, and thus, it is very efficient since $N_t \ll N_s$. On the other hand, GP_{s+t} and ATL-DGP [42] need to go through the source data in order to perform the adaptation. Hence, even with few target training data, their efficiency is bounded from the cardinality of the source domain.

Apart from the GP-based adaptation techniques, we compare to the deterministic max-margin domain transfer (MMDT) [37], that adjusts the SVM classifier to the domain adaptation scenario, and kernelized Bayesian transfer learning (KBTL) [41] that finds a shared subspace appropriate for the classification of various tasks (domains) in a probabilistic manner. Finally, we compare to state-of-the-art methods from the field of action unit analysis, *i.e.*, the dynamic SVM (dynSVM) [28] that performs the adaptation by neutral calibration (*e.g.*, removing the average, per subject, neutral image from the input data), and the confidence preserving machine

TABLE I
LEARNING AND INFERENCE COMPLEXITY OF THE GP-RELATED METHODS FOR DOMAIN ADAPTATION. THE COMPLEXITY FOR TRAINING GP_{source} IS $\mathcal{O}(N_s^3)$. GPA [24] AND THE PROPOSED (W)GPDE CAN BENEFIT FROM OFFLINE LEARNING OF THE EXPENSIVE SOURCE CLASSIFIER. FOR ATL-DGP [42] C IS THE NUMBER OF CLASSES AND M THE NUMBER OF INDUCING POINTS. NOTE THAT $N_t \ll N_s$.

	Source offline?	Adaptation	Prediction ($\boldsymbol{\mu}_*$, V_*)
GP_{source}	✗	N/A	$\mathcal{O}(N_s)$, $\mathcal{O}(N_s^2)$
GP_{s+t}	✗	$\mathcal{O}((N_s+N_t)^3)$	$\mathcal{O}(N_s+N_t)$, $\mathcal{O}((N_s+N_t)^2)$
ATL-DGP [42]	✗	$\mathcal{O}(CM^2(N_s+N_t))$	$\mathcal{O}(CM)$, $\mathcal{O}(CM^2)$
GPA [24]	✓	0	$\mathcal{O}(N_s+N_t)$, $\mathcal{O}(N_s^2+N_t^2)$
(w)GPDE	✓	$\mathcal{O}(N_t^3)$	$\mathcal{O}(N_s+N_t)$, $\mathcal{O}(N_s^2+N_t^2)$

(CPM) [25] that reweights the source classifier based on a confidence measure, before applying it to the data from the target subject. Implementations of dynSVM and CPM were not available, thus, the reported results were taken from the authors' websites. The parameters of the compared methods were tuned based on a cross-validation strategy. The proposed (w)GPDE is a non-parametric model with no free parameters to tune.

A. View adaptation from a single source: 'where'

In this experiment, we demonstrate the effectiveness of the proposed approach when the distributions between source and target domain (0°, -15° and -30°) differ in an increasing non-linear manner. For this purpose we evaluate all considered algorithms in terms of their ability to perform accurate FER as we move away from the source pose. Notice that the weighted version of our method, *i.e.*, wGPDE is not evaluated on the current experiment since FER is an intrinsic single output problem, and hence, there are no additional variances to be modeled. Furthermore, in this scenario we only considered the geometric features as inputs to the compared models since they have been proved efficient to model the global phenomena of the facial expressions [17].

Table II summarizes the results. The generic classifier GP_{source} exhibits the lowest performance, due to the fact that it has only been trained on source domain images. It is important to note the fluctuations in the classification rate when the source and target domain vary. We can clearly see that when the frontal pose, *i.e.*, 0° is used as the source domain, the symmetric nature of the face helps towards achieving a satisfactory performance on the target domains. Yet, the performance degrades when the symmetry is severely violated, *e.g.*, $0^\circ \rightarrow -30^\circ$. When -15° and -30° serve as the source domain, these symmetric attributes cannot be uncovered from the generic GP_{source}. Hence, we observe a significantly lower performance for the target frontal view (around 55%). The above results clearly indicate the inefficiency of a generic classifier to deal with data of different characteristics.

On the other hand, the GP_{target} when trained with as few as 30–50 data points, in most of the cases, achieves similar performance to the GP_{source} since it benefits from modeling domain-specific attributes. A further increase in the cardinality of the target training data results in a significant improvement in the classification rate. This is even more pronounced in the scenario we have illustrated above, *i.e.*, the target frontal view. As we can see the generic classifier when trained on the 0° can reach the CR of 84.06%, compared to the achieved 53.82% and 56.56% when trained on -15° and -30°, respectively.

The performance of the concatenated model, *i.e.*, GP_{s+t} is influenced from both the source and the target data, as was expected. When we have access to only few training target data, GP_{s+t} is influenced more from the source domain. Hence, in situations where GP_{source} performs poorly, we observe a negative transfer, and thus, GP_{s+t} cannot reach the performance of the target classifier, even with the inclusion of more target data. On the contrary, when both GP_{source} and GP_{target} achieve high performance, the GP_{s+t} manages to surpass both of them.

A similar trend can be observed in the performance of the adaptation methods, where the inclusion of 10–30 labeled data points from the target domain is adequate to shift the learned source classifier towards the distribution of the target data. The GPA uses the extra data to condition on the generic classifier GP_{source} and increase its prediction performance. Thus, it can reach its highest performance in situations where the generic classifier GP_{source} is already sufficient for the FER task (*i.e.*, -15° and -30°). However, in most cases it cannot achieve higher performance than the GP_{s+t}. This is expected since the latter learns the hyper-parameters on the concatenation of both source and target domains. On the contrary, GPA performs inference with the parameters learned using only the data from the source domain. ATL-DGP on the other hand follows the learning strategy of the GP_{s+t}, since it facilitates a joint learning scheme where GP_{source} and GP_{target} are fused together in an intermediate latent space, via conditioning, in a deep architecture. The advantage of the latter is evidenced by the highest achieved accuracy in the situations where the source classifier performs averagely, *i.e.*, $0^\circ \rightarrow -30^\circ$, $-15^\circ \rightarrow 0^\circ$ and $-30^\circ \rightarrow 0^\circ$ for $N_t = 10$–50. However, the joint training scheme of ATL-DGP limits its adaptation ability, due to the high effect of the source prior. A further disadvantage of ATL-DGP's joint learning is that it requires retraining of both source and target classifiers every time the target distribution changes.

An opposite pattern (compared to ATL-DGP) can be observed in the performance of both MMDT and KBTL. Both of these methods achieve, to some extent, to reach the accuracy of the generic GP_{target} classifier, when more and more target data become available. On the contrary their performance is problematic when dealing with quite few labeled target data, *i.e.*, $N_t < 50$. In such cases, the parametric nature of MMDT does not allow for effective learning of the projections from the target to the source domain, and hence, the learned classifier fails to poor results. Similarly, KBTL cannot recover accurate projections from the target domain data to a low-dimensional space. The latter has a negative impact on the accuracy of KBTL.

Finally, the proposed GPDE, exhibits the most stable performance for varying cardinality of labeled target data. This can be attributed to the fact that it uses the notion of experts to unify GP_{source} and GP_{target} into a single classifier. To achieve so, GPDE measures the confidence of the predictions from each expert (by means of predictive

TABLE II
AVERAGE CLASSIFICATION RATE ACROSS 5-FOLDS ON MULTIPIE. THE VIEW ADAPTATION IS PERFORMED WITH INCREASING CARDINALITY OF LABELED TARGET DOMAIN DATA ($10 - 1200$).

	Target	$-15°$								$-30°$							
	N_t	10	30	50	100	200	300	600	1200	10	30	50	100	200	300	600	1200
Source $0°$	GP_{source}	81.65								76.94							
	GP_{target}	55.85	81.19	84.59	89.61	90.66	91.31	91.57	97.26	51.99	76.09	81.97	86.48	88.57	89.75	**92.16**	**98.43**
	GP_{s+t}	82.41	84.00	85.37	88.70	90.20	91.44	94.32	96.73	77.45	79.75	81.65	85.50	87.72	87.52	89.22	94.64
	GPA [24]	82.36	84.00	85.37	88.63	90.20	91.51	93.79	96.15	77.73	79.82	81.65	85.43	87.79	87.72	89.29	93.01
	ATL-DGP [42]	**83.32**	86.34	85.22	85.62	88.16	89.82	91.24	93.72	**79.82**	**82.93**	83.36	85.53	85.63	87.41	89.17	93.91
	MMDT [37]	21.75	66.88	82.63	88.11	89.81	91.25	90.73	90.46	27.37	71.39	80.47	86.48	87.59	88.70	89.16	90.53
	KBTL [41]	41.67	69.11	72.57	85.63	87.98	89.61	91.18	97.19	34.36	62.44	66.62	81.71	84.91	86.35	89.55	95.62
	GPDE	82.95	**86.35**	**87.52**	**92.10**	**93.73**	**94.64**	**95.36**	**97.84**	78.71	82.17	**84.65**	**87.85**	**88.83**	**90.01**	91.38	96.86
	Target	$0°$								$-30°$							
	N_t	10	30	50	100	200	300	600	1200	10	30	50	100	200	300	600	1200
Source $-15°$	GP_{source}	53.82								85.70							
	GP_{target}	52.91	61.27	64.60	71.96	**77.53**	**79.10**	**81.84**	**84.06**	51.99	76.09	81.97	86.48	88.57	89.75	92.16	**98.43**
	GP_{s+t}	53.11	57.81	60.16	63.81	67.15	69.56	75.24	80.67	84.36	92.62	93.21	93.75	94.53	**95.89**	**96.02**	98.01
	GPA [24]	55.00	57.67	59.70	63.10	65.51	68.26	72.83	78.31	88.37	92.16	93.21	93.86	94.45	94.97	95.30	97.52
	ATL-DGP [42]	**70.11**	**73.20**	**71.15**	72.21	73.48	75.87	79.91	82.15	78.33	79.95	82.68	85.12	86.79	89.44	91.76	95.33
	MMDT [37]	17.37	42.91	63.03	71.72	72.44	74.98	78.18	79.23	11.93	63.10	86.54	90.27	89.55	90.40	89.03	86.81
	KBTL [41]	22.08	35.99	59.24	67.28	70.35	71.39	75.11	79.03	32.20	64.21	70.35	82.89	87.00	87.85	90.73	96.41
	GPDE	56.11	63.23	66.82	**72.37**	75.64	76.94	80.40	83.80	**88.44**	**93.40**	**94.32**	**93.99**	**94.84**	94.64	94.97	98.04
	Target	$0°$								$-15°$							
	N_t	10	30	50	100	200	300	600	1200	10	30	50	100	200	300	600	1200
Source $-30°$	GP_{source}	56.56								91.38							
	GP_{target}	52.91	61.27	64.60	71.96	**77.53**	**79.10**	**81.84**	**84.06**	55.85	81.19	84.59	89.61	90.66	91.31	91.57	97.26
	GP_{s+t}	57.22	60.42	61.59	65.38	67.34	70.02	75.70	80.67	92.68	94.51	94.81	95.75	**96.21**	**97.06**	**96.93**	98.24
	GPA [24]	57.41	59.83	61.53	64.53	67.15	69.24	75.11	77.60	93.27	94.58	94.72	95.43	95.89	96.54	96.47	97.91
	ATL-DGP [42]	**70.13**	**75.38**	**73.45**	**74.79**	74.68	77.23	79.92	82.03	83.52	84.21	84.94	85.02	87.90	89.84	92.13	94.63
	MMDT [37]	20.77	46.11	60.81	69.76	72.63	76.55	78.71	79.69	23.97	72.11	86.41	92.36	92.36	92.68	93.08	92.42
	KBTL [41]	22.08	35.60	59.37	67.60	70.15	71.06	74.85	78.18	40.10	68.26	75.38	87.72	89.42	90.01	91.70	97.58
	GPDE	59.57	65.58	69.56	72.57	75.96	77.86	81.45	83.61	**93.60**	**94.64**	**94.84**	**94.58**	94.51	94.25	93.60	**98.37**

variance), in contrast to GPA (uses source expert only) and ATL-DGP (uses an uninformative prior). This property of GPDE is more pronounced in the highly non-linear adaptation scenarios of $0° \rightarrow -30°$, $-30° \rightarrow 0°$ and $-15° \rightarrow 0°$ for $N_t > 200$, where GP_{target} achieves the highest classification ratio. GPDE performs similarly to the target expert while, GPA and ATL-DGP underestimate the prediction capacity of the target-specific classifier, and thus, attain lower results. The only situations where GPDE achieves inferior performance are the cases where GP_{source} performs poorly. Thus, as expected, GPDE cannot attain a reliable adaptation without having access to latent factors, opposed to ATL-DGP.

B. Subject adaptation from multiple sources: 'who'

In this section, we evaluate the models in a multi-label classification scenario, where the adaptation is performed from multiple source domains. This is also a natural setting to demonstrate the importance of modeling different variances per output dimensions with the proposed wGPDE. In contrast to the view adaptation scenario for FER, herein we report results for both geometric and appearance features, since different AUs are better explained from different type of features.

Overall, this is a more challenging setting, since the datasets are comprised of naturalistic facial expressions, and the recorded subjects are experiencing the affect in different ways and levels. The difficulty of the task can be seen in Fig. 3, where the subject-specific classifier GP_{target} trained with 10–30 labeled data points, achieves a higher average F1 score than the generic classifier GP_{source}, which is trained on all available source subjects. The importance of this outcome gets more clear if we consider that it holds for both DISFA and FERA2015, when using either geometric or appearance features. This suggests that, no matter the nature of the inputs, personalized AU detectors are superior to generic classifiers, even when limited data are available. Another factor that is worth mentioning is that the average results are obtained over a large set of AUs (*i.e.*, 12 AUs for DISFA and 11 AUs for FERA2015). This fact, not only constitutes the results more reliable, but it also implies that even a small increase in the average performance (*e.g.*, 1-2%) can be attributed to an improved performance over several AUs.

By continuing our analysis of Fig. 3 we observe that the adaptation models, *i.e.*, GPA, GPDE and wGPDE achieve superior F1 score compared to the generic GP_{target}, under all scenarios. The latter implies that images from source and target subjects contain complementary information regarding the depicted facial expressions. Hence, the target classifier does not consist anymore an upper bound limit for the adaptation. This can be explained from the multi-modal nature of the problem , since we can have different AU combinations per sequence, contrary to the universal expressions appearing in the view adaptation scenario. Thus, expressions that are present only on the source sequences, can be used to improve the AU detection task for the target subject. Note also that the classifier trained on the concatenation of the source and target domains, *i.e.*, GP_{s+t}, outperforms almost all models on DISFA. However, this is not the case on FERA2015 dataset, where the subject differences are more pronounced due to the high resolution images. Hence, GP_{s+t} fails to the performance achieved by either GP_{source} or GP_{target} classifier. The proposed GPDE and wGPDE benefit from modeling the target-specific information and can attain

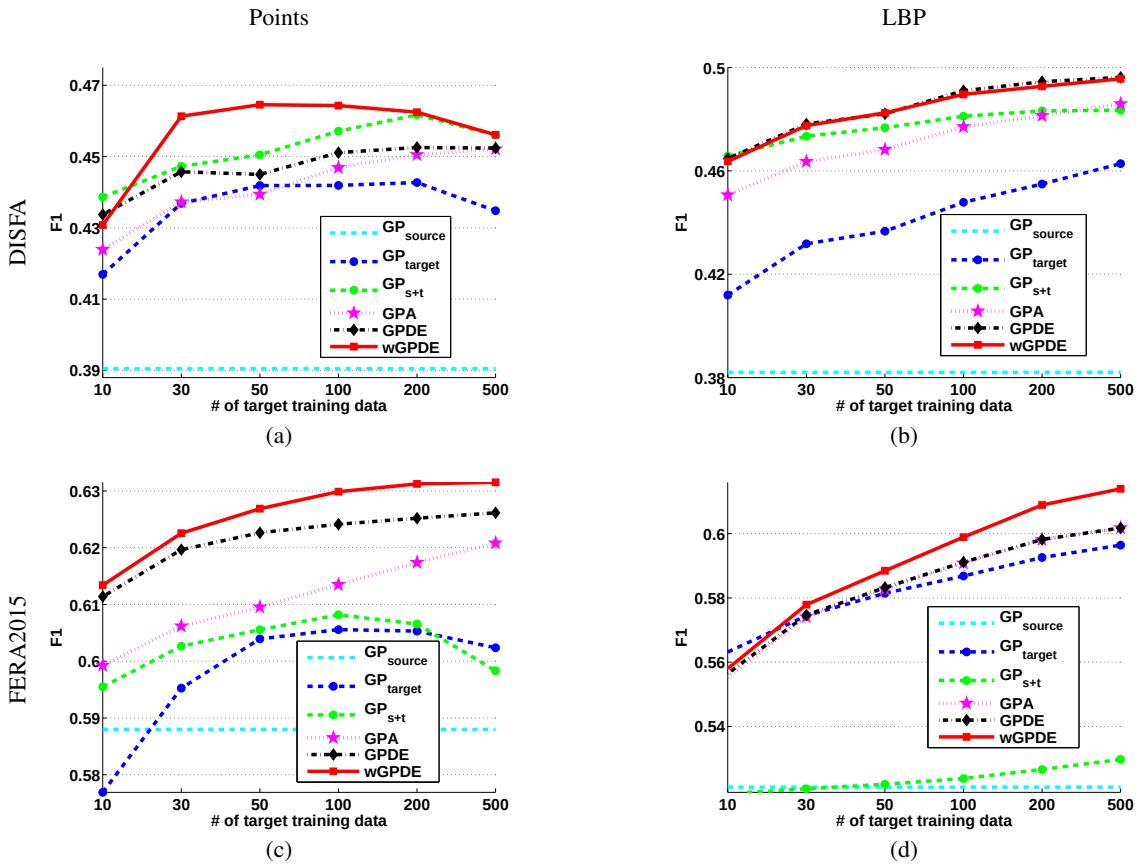


Fig. 3. Average F1 score for joint AU detection with subject adaptation on DISFA (top) and FERA2015 (bottom) with increasing number of target domain data. The results are reported when using geometric (left) and appearance (right) features.

a better adaptation compared to GPA. Another reason for the difference in the performance between the proposed model and GPA is that the latter treats all training subjects as data from a single, *broader*, source domain. Hence, GPA smooths out the individual differences and lessens the contribution of the target domain, as the variations of the target data can be explained, on average, by the source domain.

Finally, the importance of modeling individual variances becomes evident by comparing the attained scores from wGPDE and GPDE. In 3 out of 4 scenarios, wGPDE achieves superior performance with more pronounced results on DISFA dataset when geometric features are used (see Fig. 3(a)). On the other hand, when appearance features are used, as we can see in Fig. 3(b) both wGPDE and GPDE perform similarly. This can be explained by the fact that images from DISFA are not of high resolution. Hence, the local patches cannot explain adequately all the important variations that differ among the various outputs (*i.e.*, AUs). However, as we can see in Fig. 3(d) this is not the case with the high-resolution images from FERA2015. The input appearance features are of better quality, and thus, wGPDE can more accurately model the individual variances per output, and thus, attain higher scores.

For a better understanding of the efficacy of the adaptation task, in Table III we report the detailed results per AU for the case of $N_t = 50$. Note that this setting is not always the most beneficial for our proposed approach. In most scenarios the gap in the performance between (w)GPDE and the other methods increases as we include more target data. However, we demonstrate the performance on $N_t = 50$ because AU annotations are expensive and laborious. Thus, such a setting is a more reasonable choice for adaptation for the current task. The proposed (w)GPDE under the current setting, and using the geometric features as input (upper half of Table III), attains an average F1 improvement on both DISFA and FERA2015 of 2%. This small increase in the average performance translates to an improved F1 score on 6/12 and 8/11 AUs, respectively. The robustness of (w)GPDE is further supported by both per AU and average AUC. We can see that (w)GPDE achieves higher AUC even in the AUs that reports inferior F1 score, resulting in 9/12 and 10/11 improved AUs on DISFA and FERA2015, respectively. Thus, it is evident that (w)GPDE constitutes a more reliable classifier, under these settings. Regarding the appearance features (lower half of Table III) the average improvement of (w)GPDE is marginal, especially on FERA2015. Yet, if we look again individually at each AU, we observe that the proposed model attains increased F1 score on 6/12 (8/12 in terms of AUC) and 7/11 (11/11 in terms of AUC), on DISFA and FERA2015, respectively.

By comparing wGPDE to GPDE we can further observe that modeling of individual variances results in improved average performance, which translates to an improvement on certain AUs. An indicative example is the increase in F1 of AUs $1, 2, 5, 6$ on DISFA, especially when using the geometric features. On all these 4 AUs, the standard GPDE fails to

TABLE III
F1 SCORE AND AUC FOR JOINT AU DETECTION ON DISFA AND FERA2015. SUBJECT ADAPTATION WITH $N_t = 50$.

		Dataset	DISFA													FERA2015											
		AU	1	2	4	5	6	9	12	15	17	20	25	26	Avg.	1	2	4	6	7	10	12	14	15	17	23	Avg.
Points	F1	GP_{source}	33.1	31.6	54.8	10.5	44.8	31.6	57.3	24.4	35.8	13.7	**79.5**	51.5	39.0	49.5	34.5	57.9	73.9	77.2	79.5	82.2	62.6	32.1	60.2	37.2	58.8
		GP_{target}	37.2	41.4	62.2	21.7	57.3	30.2	59.3	25.9	38.3	20.5	76.0	60.1	44.2	43.4	38.5	53.3	72.2	78.3	83.7	80.7	64.6	**48.5**	60.8	41.0	60.5
		GP_{s+t}	**42.1**	48.3	61.1	19.2	45.2	**42.1**	63.1	23.8	**41.0**	**23.9**	76.2	54.6	45.1	52.9	37.3	59.4	74.1	77.7	81.5	82.1	64.4	34.9	61.5	40.2	60.6
		GPA [24]	36.0	37.2	62.4	21.3	52.7	36.4	**67.3**	**27.1**	38.7	16.2	77.1	54.8	43.9	**54.6**	37.8	**60.4**	74.9	77.9	81.5	83.1	64.6	34.7	61.4	39.7	61.0
		GPDE	36.8	38.3	**63.2**	22.7	54.3	36.8	66.4	26.8	38.9	16.5	77.4	55.9	44.5	52.6	38.8	57.8	**75.7**	**79.2**	**84.9**	**84.5**	**65.9**	39.1	65.2	40.7	62.3
		wGPDE	41.2	**52.9**	61.7	**25.3**	**60.9**	32.8	58.8	**27.1**	40.7	16.7	77.6	**65.2**	**46.8**	53.4	**41.2**	58.5	75.1	79.0	84.2	83.4	65.6	40.9	**65.7**	**43.1**	**62.7**
	AUC	GP_{source}	71.3	73.2	64.1	56.3	70.7	71.8	77.3	61.6	65.7	57.4	80.2	67.7	68.2	75.5	65.9	81.5	81.5	68.9	76.1	85.9	66.7	57.5	68.5	65.6	72.1
		GP_{target}	72.6	77.2	75.2	63.3	81.6	66.8	75.7	61.3	69.0	69.3	77.8	74.3	72.0	67.6	68.9	77.0	76.5	73.1	82.6	79.1	70.8	73.2	68.6	68.1	73.2
		GP_{s+t}	74.7	75.0	75.4	68.6	73.0	**78.9**	80.2	61.0	**72.9**	**71.5**	78.7	72.3	73.5	76.9	67.9	82.7	81.7	70.8	81.3	85.5	70.1	63.6	69.9	68.5	74.5
		GPA [24]	74.9	76.8	75.3	68.1	79.9	73.7	81.2	66.3	71.1	63.1	79.7	73.6	73.6	**79.1**	68.7	83.4	83.0	72.2	81.4	87.1	70.1	63.3	69.8	68.5	75.1
		GPDE	**75.5**	77.6	**76.2**	68.3	81.2	73.9	**81.3**	**66.4**	71.5	63.8	80.3	74.6	74.2	72.7	69.3	83.2	**83.3**	**76.7**	85.5	**88.4**	**73.7**	68.6	**75.2**	70.5	77.0
		wGPDE	73.7	**83.2**	75.0	**71.4**	**82.9**	72.3	77.0	64.2	70.6	60.8	**80.4**	**79.4**	**74.3**	74.1	**70.6**	**83.6**	82.7	76.6	**85.6**	87.6	**73.7**	**71.0**	74.9	**72.2**	**77.5**
LBP	F1	GP_{source}	31.0	27.0	52.2	11.7	35.5	29.3	52.4	31.1	38.6	23.8	73.4	52.4	38.2	35.8	29.9	36.0	63.3	75.8	78.1	73.1	60.5	30.6	58.0	32.1	52.1
		GP_{target}	35.4	40.9	58.7	10.5	55.4	30.6	56.2	28.9	40.7	23.0	79.7	64.1	43.7	**41.6**	36.4	48.1	64.9	**78.0**	80.9	74.7	63.0	**50.0**	58.8	43.2	58.1
		GP_{s+t}	39.9	39.3	63.2	**28.8**	48.8	**38.7**	**66.4**	**34.9**	**47.4**	**26.4**	81.4	56.6	47.7	35.0	28.6	36.7	63.3	75.7	78.1	73.2	60.6	30.6	58.0	34.2	52.2
		GPA [24]	38.5	37.3	63.4	13.6	62.0	32.4	63.8	30.9	44.9	24.4	83.1	67.7	46.8	41.2	36.5	46.8	66.9	77.4	80.3	76.8	62.6	47.6	60.1	**44.7**	58.3
		GPDE	39.8	41.1	65.1	17.2	**62.2**	34.5	64.3	32.5	44.9	25.5	**83.4**	**68.2**	**48.2**	41.4	36.6	47.0	66.8	77.4	80.5	76.7	62.6	47.7	60.1	44.7	58.3
		wGPDE	**41.0**	**41.8**	**65.6**	20.8	60.7	34.1	60.9	34.5	46.3	24.4	82.1	66.7	**48.2**	41.4	**37.3**	**48.7**	**68.6**	77.6	**81.6**	**77.6**	**63.2**	47.4	**60.6**	44.4	**58.9**
	AUC	GP_{source}	67.2	66.4	57.3	66.3	60.2	68.7	69.7	68.6	69.4	73.6	75.2	68.7	67.6	56.3	58.5	54.0	41.5	47.2	40.4	42.3	47.8	51.5	47.5	55.3	49.3
		GP_{target}	75.8	77.9	71.1	60.8	81.3	71.8	75.0	68.3	72.1	71.5	84.0	80.4	74.2	65.4	65.3	72.3	62.6	71.5	75.0	63.5	68.6	76.0	62.8	71.0	68.5
		GP_{s+t}	76.4	74.2	77.3	**81.2**	76.9	**76.7**	**85.2**	74.6	**78.0**	74.1	87.6	73.7	78.0	53.9	56.2	58.7	55.6	53.9	56.3	58.3	52.6	54.0	53.1	59.6	55.7
		GPA [24]	78.3	80.0	77.5	70.2	84.4	73.2	81.4	72.1	75.4	74.9	88.2	83.0	78.2	66.8	65.9	72.6	71.1	73.1	77.6	74.2	69.5	74.0	65.5	72.0	71.1
		GPDE	79.7	**82.2**	79.6	76.1	**84.5**	75.2	82.3	74.6	75.4	**75.3**	**88.5**	**83.4**	79.7	**66.9**	66.0	72.6	70.8	73.2	77.6	73.8	69.6	74.2	65.4	**72.1**	71.1
		wGPDE	**80.4**	82.1	**81.0**	79.4	83.7	75.3	80.2	**76.1**	76.0	73.9	87.4	82.0	**79.8**	65.9	**66.6**	**74.7**	**74.7**	**73.6**	**79.6**	**77.4**	**70.3**	**73.9**	**66.9**	71.9	**72.3**

reach the performance of the generic GP_{target} classifier. However, the proposed weighting allows the GPDE to model output-specific attributes, or 'pair' the variances that are associated with co-occurring outputs, *e.g.*, AUs $1, 2$. Similar pattern can be observed in the results for AU2, for geometric, and AUs $2, 4, 6$, for appearance features on FERA2015. Especially for AUs $4, 6$ the increase in F1 is further supported by an increase in AUC of 2% and 4%, respectively.

We next compare the proposed (w)GPDE to the state-of-the-art models from the literature on AU analysis, which attempt to perform the adaptation. We compare to the supervised dynSVM [28] and the semi-supervised CPM [25]. dynSVM attempts to perform the adaptation at the feature level (combination of geometric and appearance features), where the input data from each subject (domain) are normalized by removing the dynamics of the expression. CPM on the other hand tries to adjust the classifier to the target domain. It achieves so by taking into account the confidence/agreement in the predictions of source soft classifiers, when assessing the target data.

Table IV summarizes the results. At first we can see that the proposed wGPDE outperforms both dynSVM and CPM on both DISFA and FERA2015. The improvement over dynSVM on DISFA is marginal. However, the authors in [28], before applying the dynSVM, attempted to re-balance the data in order to account for the mismatch in the distribution of activated AUs. This explains the superior performance of dynSVM on less frequently occurring AUs, *i.e.*, AUs $9, 15, 20$ on DISFA and AUs $14, 23$ on FERA2015. On the other hand, CPM reports lower results, both on average and per AU, on both datasets. This is partly attributed to the fact that CPM is a semi-supervised method and uses soft labels (*i.e.*, the predictions of the source classifier) as ground truth labels for the target data during training. Another reason for its low performance is the 'virtual' way that CPM utilizes to measure the confidence. In contrast, the proposed wGPDE has a well determined probabilistic way to correctly estimate the confidence in the predictions of the various experts. This allows the wGPDE to weight the contribution of each expert in the final classification, which results in more accurate predictions.

C. Assessing the confidence in the predictions

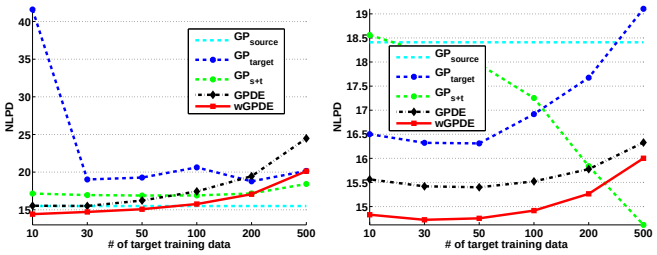


Fig. 4. Quantification of the confidence in the probabilistic predictions in terms of NLPD for DISFA (left) and FERA2015 (right) with increasing number of target domain data.

Herein, we assess the ability of (w)GPDE to measure the confidence in the output labels, by means of the predicted variance. To this end, we use the negative log-predictive density (NLPD) as an evaluation measure. It is commonly used in probabilistic models, since it takes into account the predictive variance. In Fig. 4 we see the NLPD for the baseline generic classifiers, *i.e.*, GP_{source}, GP_{target} and GP_{s+t}, and for the proposed (w)GPDE, on both DISFA and FERA2015 datasets. First of all we observe that all the models (apart from the GP_{target} on DISFA and GP_{s+t} on FERA2015) increase their variance in the predictions (NLPD is increasing), as we include more training target data. This, however, is expected since by increasing the training set, we observe more variations in the input data (different AU combinations). Hence, the variance in the outputs also increases. In the case of DISFA, (Fig. 4(left)) the target expert becomes more confident for $N_t > 10$. We attribute this to the nature of the videos in DISFA, which contain less frequently varying expressions over

TABLE IV
F1 SCORE FOR JOINT AU DETECTION ON DISFA AND FERA2015. COMPARISON TO STATE-OF-THE-ART. SUBJECT ADAPTATION FOR WGPDE HAS BEEN PERFORMED WITH $N_t = 50$.

Dataset	DISFA													FERA2015										
AU	1	2	4	5	6	9	12	15	17	20	25	26	Avg.	1	2	4	6	7	10	12	14	15	17	23	Avg.
wGPDE (pts.)	**41.2**	**52.9**	61.7	**25.3**	**60.9**	32.8	58.8	27.1	40.7	16.7	77.6	65.2	46.8	**53.4**	**41.2**	**58.5**	75.1	**79.0**	84.2	83.4	65.6	40.9	**65.7**	43.1	**62.7**
wGPDE (app.)	41.0	41.8	**65.6**	20.8	60.7	34.1	60.9	34.5	**46.3**	24.4	82.1	66.7	**48.2**	41.4	37.3	48.7	68.6	77.6	81.6	77.6	63.2	**47.4**	60.6	44.4	58.9
dynSVM [28]	30.0	26.0	34.0	16.0	45.0	**45.0**	**77.0**	**47.0**	41.0	**25.0**	**84.0**	**75.0**	48.0	43.0	39.0	46.0	**77.0**	77.0	**85.0**	**87.0**	**67.0**	44.0	62.0	**45.0**	61.0
CPM [25]	29.5	24.8	56.8	–	41.7	31.5	71.9	–	–	–	81.6	51.3	–	46.6	38.7	46.5	68.4	73.8	74.1	84.6	62.2	44.3	57.5	41.7	58.0

time. Thus, the generic personalized classifier has seen most of the available variations – on average – which results in reduced uncertainty. On the other hand, the events on FERA2015 are shorter, hence, more frequent variations. Thus, the relevant NLPD at first decreases, but as more data become available (more AU combinations) the uncertainty increases. Eventually, in both situations the generic GP_{target} becomes less confident than GP_{source}. In contrast, this is not the case for the GP_{s+t} classifier on FERA2015. The weird behavior of GP_{s+t} is an indication that it focuses on universal characteristics and variations on the face, which are irrelevant to the task of AU detection. Hence, the more data it sees, the more confident it becomes, yet it still predicts with low F1 score, as can be also seen from Fig. 3(d).

By comparing GPDE to wGPDE, we observe a similar modeling behavior. However, GPDE without the weighting can only produce a single variance for all outputs. This has a negative impact on the NLPD, since the model is equally confident for all the outputs. Thus, GPDE results in being over-confident, even for false predictions. On the other hand, the weighting term allows the wGPDE to produce different variance for each predicted output.

The above claims for the difference between GPDE and wGPDE are better explained from Fig. 5. In Fig. 5(top) we see an example where both GPDE and wGPDE predict the exact same labels (almost the same predicted means). However, GPDE (Fig. 5(left)) suffers from heavier tails. This results in less accurate estimation of the mass probability for AUs $1, 2, 10, 12$, which can be interpreted by also a higher NLPD. The same behavior of heavier tails can be observed in another example in Fig. 5(bottom). However, now GPDE and wGPDE disagree on their predictions for AUs $6, 17$. wGPDE can better estimate the probability mass for the quite uncertain AUs $6, 17$, which results in their correct prediction compared to the unweighted GPDE.

D. Cross dataset adaptation

Herein, we evaluate the robustness of the models when performing the subject adaptation, in a more challenging scenario. We perform two different cross-dataset experiments, FERA2015→DISFA and DISFA→FERA2015.[2] Note that if the same subjects were present on both datasets we could also address the question 'what', by modeling the causal factor that elicited the depicted facial expressions across the datasets. Since we lack the appropriate data, we focus only on the question 'who'. We evaluate the models' performance on 7 AUs (*i.e.*, $1, 2, 4, 6, 12, 15, 17$) that are present in both datasets. We employ the geometric features, since the images from the two datasets differ significantly in resolution. However, even the geometric features are being affected by factors, such as, facial pose and size. This imposes a further difficulty on the alignment of the input facial features.

By analyzing the results in Fig. 6 we can draw two quick conclusions. First, FERA2015 is a more representative dataset for the task of AU detection. The generic classifier GP_{source} in Fig. 6(left) achieves similar performance to the adaptation models in Fig. 3(a). This does not hold for the generic GP_{source} in the DISFA→FERA2015 experiment. The latter is further supported by the performance of GP_{target} and GP_{s+t}, which by including information from the target data they can significantly outperform the generic GP_{source} on the DISFA→FERA2015 adaptation. The second finding is related to the advantage of the joint modeling of the AUs. This is illustrated in the performance of the generic GP_{target} in both cross-dataset evaluations. We can see that the average results are lower than the corresponding ones from Table III.

Regarding the performance of the adaptation methods we observe that in the FERA2015→DISFA scenario, all the compared models benefit from the presence of the additional target domain data. More interestingly, (w)GPDE consistently outperforms GPA and reaches the average performance of the corresponding AUs in the within dataset evaluations from Table III. The importance of wGPDE is not obvious in this scenario. However, in the DISFA→FERA2015 adaptation, wGPDE manages to correctly model the individual variances in the target data, and hence, achieves better performance than the generic GP_{target} (contrary to the simple GPDE).

Finally, the detailed results per AU for the cross dataset adaptation are presented in Table V. It is clear that the proposed approach, not only outperforms its counterparts on the current experiment, but also achieves improved performance on most of the AUs (particularly in FERA2015→DISFA), compared to the within dataset evaluations. This is an indicator of the quality of the achieved adaptation, since the model becomes less sensitive to the input source data. On the other hand, the subject normalization of dynSVM does not attain a sufficient adaptation, and hence, it fails to lower results than the generic GP_{source}.

VI. DISCUSSION AND CONCLUSIONS

From the conducted experiments on various adaptation scenarios, we made several important observations: the source classifier trained on a large number of data can easily be outperformed by the classifier trained on as few as 50 examples

[2] 'A→B' denotes the training on dataset A and testing on dataset B.

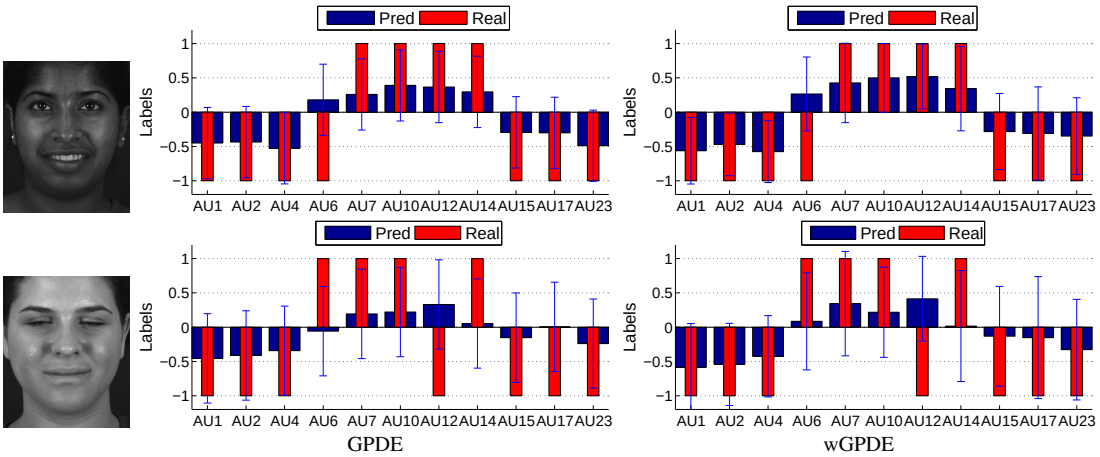
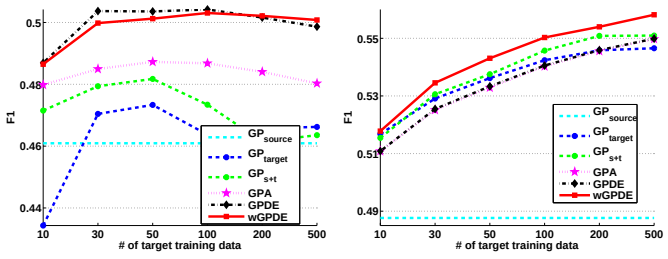


Fig. 5. Probabilistic prediction of joint AU activations on FERA2015 from GPDE (left) and wGPDE (right). The reported tails account for the predicted standard deviation. Shorter tails correspond to more confident predictions. Both GPDE and wGPDE are trained with $N_t = 50$.

TABLE V
CROSS-DATASET EVALUATIONS ON 7 AUS PRESENT IN BOTH DISFA AND FERA2015 DATASETS. THE MODELS ARE TRAINED ON FERA2015 AND TESTED ON DISFA DATASET (F → D), AND THE OTHER WAY AROUND (D → F). SUBJECT ADAPTATION WITH $N_t = 50$.

	AU	F1 1	2	4	6	12	15	17	Avg.	AUC 1	2	4	6	12	15	17	Avg.
F → D	GP_{source}	**44.0**	43.9	56.4	49.1	54.8	28.9	45.6	46.1	77.3	81.0	65.2	73.7	72.5	66.4	**75.4**	73.1
	GP_{target}	39.2	46.4	58.2	61.0	57.3	**29.6**	39.7	47.3	74.4	81.8	70.8	81.1	73.0	65.8	68.0	73.6
	GP_{s+t}	44.3	45.7	59.1	55.6	59.9	27.7	44.9	48.2	78.1	**82.6**	71.8	81.9	77.5	65.7	**75.4**	76.2
	GPA [24]	41.3	44.7	61.9	57.2	62.9	28.7	44.4	48.7	78.3	80.7	74.6	82.0	79.4	67.6	73.5	76.6
	dynSVM [28]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
	GPDE	41.8	44.8	**63.9**	**61.7**	**66.5**	28.1	**45.8**	**50.4**	79.1	81.9	**76.5**	**85.0**	**82.4**	67.6	75.1	**78.2**
	wGPDE	43.4	**46.9**	62.4	61.5	63.9	**29.6**	43.2	50.1	**80.4**	81.7	75.1	84.5	80.3	**68.6**	73.2	77.7
D → F	GP_{source}	37.3	28.0	46.5	63.8	74.1	31.6	60.1	48.8	61.1	55.5	71.7	64.8	74.9	50.9	61.9	63.0
	GP_{target}	41.1	**37.5**	47.0	67.5	77.0	**45.8**	59.4	53.6	67.0	66.4	71.7	68.1	69.3	71.1	63.7	68.2
	GP_{s+t}	**47.1**	**37.5**	52.8	67.5	77.6	34.0	59.8	53.8	**71.6**	**67.9**	77.9	73.9	80.7	61.5	67.6	71.6
	GPA [24]	40.7	36.3	50.6	68.0	76.9	39.7	60.8	53.3	67.3	65.2	74.6	72.8	76.0	69.0	66.2	70.2
	dynSVM [28]	44.0	34.0	50.0	68.0	67.0	26.0	48.0	48.0	–	–	–	–	–	–	–	–
	GPDE	40.7	36.4	50.5	68.0	77.0	40.0	60.7	53.3	67.3	65.3	74.6	72.7	75.8	**69.2**	66.2	70.2
	wGPDE	**42.1**	35.9	**54.7**	**69.2**	**79.5**	36.9	**62.0**	**54.3**	66.3	64.3	**79.5**	**76.5**	**83.6**	66.5	**69.6**	**72.3**



FERA2015 → DISFA DISFA→ FERA2015

Fig. 6. Cross-dataset evaluations. Average F1 score of the 7 common AUs present in both DISFA and FERA2015 datasets. The models are trained on data from FERA2015 and tested on data from DISFA (left), and the other way around (right). The reported results are obtained with geometric features and increasing cardinality of labeled target domain data.

from the target domain. Furthermore, the existing adaptation approaches try to adapt the target domain to the source domain by assuming that the two distributions can be matched. Yet, when more target data become available, a generic target classifier can largely outperform the existing adaptation approaches. To address the aforementioned challenges, we have presented a method that exploits successfully the non-parametric probabilistic framework of GPs to perform domain adaptation for both multi-class and multi-label classification of human facial expressions. In contrast to existing adaptation approaches, which leverage solely the source distribution during adaptation, the proposed approach defines a target expert to model domain-specific attributes, and reduce that way the effect of negative transfer. As a purely probabilistic model, (w)GPDE explores also the variance in the predictions. The latter consists an accurate measure of confidence, and as such, it can be used to reevaluate the predictions from the various experts to achieve an improved classification performance.

To conclude, in the current work we demonstrated the advantages of the proposed (w)GPDE by performing adaptation of two contextual factors: '*who*' (subject) and '*where*' (view). In our future work we plan to explore the remaining contextual factors (*i.e.*, 'when', 'why', 'what' and 'how'), simultaneously to attain a general framework for adaptation. Although the 'when' and 'how' factors can easily be incorporated in our framework, by accounting for the temporal and multi-modal (*e.g.*, video and audio) information in the sequences, respectively, adaptation of the other factors is more difficult, especially due to the lack of appropriate data.

ACKNOWLEDGMENT

This work has been funded by the European Community Horizon (H2020) under grant agreement No. 645094 (SEWA), and No. 688835 (DE-ENIGMA). The work of O. Rudovic has also in part been

supported by H2020 research program under the Marie Skłodowska-Curie grant agreement No. 701236 (EngageME). M. P. Deisenroth has been supported by a Google Faculty Research Award.

REFERENCES

[1] N. Ambady and R. Rosenthal, "Thin slices of expressive behavior as predictors of interpersonal consequences: A meta-analysis." *APA Psychological Bulletin*, vol. 111, no. 2, p. 256, 1992.

[2] Z. Zeng, M. Pantic, G. Roisman, and T. Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 39–58, 2009.

[3] Y.-L. Tian, T. Kanade, and J. F. Cohn, "Facial expression analysis," in *Handbook of face recognition*, 2005, pp. 247–275.

[4] J. F. Cohn and P. Ekman, "Measuring facial action," *The new handbook of methods in nonverbal behavior research*, pp. 9–64, 2005.

[5] P. Ekman, W. V. Friesen, and J. C. Hager, "Facial action coding system," *Salt Lake City, UT: A Human Face*, 2002.

[6] M. Pantic, "Machine analysis of facial behaviour: Naturalistic and dynamic behaviour," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 364, no. 1535, pp. 3505–3513, 2009.

[7] J. M. Girard, J. F. Cohn, L. A. Jeni, S. Lucey, and F. D. la Torre, "How much training data for facial action unit detection?" in *IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, 2015.

[8] O. Rudovic, V. Pavlovic, and M. Pantic, "Context-sensitive dynamic ordinal regression for intensity estimation of facial action units," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 37, no. 5, pp. 944–958, 2015.

[9] M. F. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. Scherer, "The first facial expression recognition and analysis challenge," in *IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, 2011, pp. 921–926.

[10] W.-S. Chu, F. D. L. Torre, and J. F. Cohn, "Selective transfer machine for personalized facial action unit detection," in *IEEE Conf. on Computer Vision & Pattern Recognition*, 2013, pp. 3515–3522.

[11] E. Sangineto, G. Zen, E. Ricci, and N. Sebe, "We are not all equal: Personalizing models for facial expression analysis with transductive parameter transfer," in *ACM Conf. on Multimedia*, 2014, pp. 357–366.

[12] Z. Zhu and Q. Ji, "Robust real-time face pose and facial expression recovery," in *IEEE Conf. on Computer Vision & Pattern Recognition*, vol. 1, 2006, pp. 681–688.

[13] Y. Hu, Z. Zeng, L. Yin, X. Wei, J. Tu, and T. Huang, "A study of non-frontal-view facial expressions recognition," in *Int'l Conf. on Pattern Recognition.*, 2008, pp. 1–4.

[14] S. Moore and R. Bowden, "Local binary patterns for multi-view facial expression recognition," *Computer Vision and Image Understanding*, vol. 115, no. 4, pp. 541–558, 2011.

[15] O. Rudovic, M. Pantic, and I. Patras, "Coupled Gaussian processes for pose-invariant facial expression recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1357–1369, 2013.

[16] N. Hesse, T. Gehrig, H. Gao, and H. K. Ekenel, "Multi-view facial expression recognition using local appearance features," in *Int'l Conf. on Pattern Recognition.*, 2012, pp. 3533–3536.

[17] S. Eleftheriadis, O. Rudovic, and M. Pantic, "Discriminative shared Gaussian processes for multiview and view-invariant facial expression recognition." *IEEE Trans. on Image Processing*, vol. 24, no. 1, pp. 189–204, 2015.

[18] C. Rasmussen and C. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 1.

[19] M. P. Deisenroth and J. W. Ng, "Distributed Gaussian processes," *Int'l Conf. on Machine Learning*, 2015.

[20] Y. Cao and D. J. Fleet, "Generalized product of experts for automatic and principled fusion of Gaussian process predictions," *arXiv preprint arXiv:1410.7827*, 2014.

[21] G. Zen, E. Sangineto, E. Ricci, and N. Sebe, "Unsupervised domain adaptation for personalized facial emotion recognition," in *Int'l Conf. on Multimodal Interaction*, 2014, pp. 128–135.

[22] J. Chen, X. Liu, P. Tu, and A. Aragones, "Learning person-specific models for facial expression and action unit recognition," vol. 34, no. 15, pp. 1964–1970, 2013.

[23] Y.-Q. Miao, R. Araujo, and M. S. Kamel, "Cross-domain facial expression recognition using supervised kernel mean matching," in *Int'l Conf. on Machine Learning and Applications*, 2012, pp. 326–332.

[24] B. Liu and N. Vasconcelos, "Bayesian model adaptation for crowd counts," in *IEEE Int'l Conf. on Computer Vision*, 2015, pp. 4175–4183.

[25] J. Zeng, W.-S. Chu, F. De la Torre, J. F. Cohn, and Z. Xiong, "Confidence preserving machine for facial action unit detection," in *IEEE Int'l Conf. on Computer Vision*, 2015, pp. 3622–3630.

[26] M. Seeger, "Bayesian Gaussian process models: PAC-Bayesian generalisation error bounds and sparse approximations," Ph.D. dissertation, University of Edinburgh, 2003.

[27] S. Eleftheriadis, O. Rudovic, M. P. Deisenroth, and M. Pantic, "Gaussian process domain experts for model adaptation in facial behavior analysis," *IEEE Conf. on Computer Vision & Pattern Recognition, Workshops*, 2016.

[28] T. Baltrušaitis, M. Mahmoud, and P. Robinson, "Cross-dataset learning and person-specific normalisation for automatic action unit detection," in *IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, 2015.

[29] A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schölkopf, "Covariate shift by kernel mean matching," *Dataset shift in machine learning*, 2009.

[30] T. Almaev, B. Martinez, and M. Valstar, "Learning to transfer: transferring latent task structures and its application to person-specific facial action unit detection," in *IEEE Int'l Conf. on Computer Vision*, 2015, pp. 3774–3782.

[31] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 53–69, 2015.

[32] K. Zhang, B. Schölkopf, K. Muandet, and Z. Wang, "Domain adaptation under target and conditional shift." in *Int'l Conf. on Machine Learning*, 2013, pp. 819–827.

[33] M. Gong, K. Zhang, T. Liu, D. Tao, C. Glymour, and B. Schölkopf, "Domain adaptation with conditional transferable components," in *Int'l Conf. on Machine Learning*, 2016, pp. 2839–2848.

[34] H. Daumé III, "Frustratingly easy domain adaptation," *Trans. of the Association for Computational Linguistics*, p. 256, 2007.

[35] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms," in *IEEE Conf. on Computer Vision & Pattern Recognition*, 2011, pp. 1785–1792.

[36] J. Hoffman, B. Kulis, T. Darrell, and K. Saenko, "Discovering latent domains for multisource domain adaptation," in *Eur. Conf. on Computer Vision*, 2012, pp. 702–715.

[37] J. Hoffman, E. Rodner, J. Donahue, K. Saenko, and T. Darrell, "Efficient learning of domain-invariant image representations," in *Int'l Conf. on Learning Representations*, 2013.

[38] J. Donahue, J. Hoffman, E. Rodner, K. Saenko, and T. Darrell, "Semi-supervised domain adaptation with instance constraints," in *IEEE Conf. on Computer Vision & Pattern Recognition*, 2013, pp. 668–675.

[39] L. Duan, D. Xu, and I. Tsang, "Learning with augmented features for heterogeneous domain adaptation," in *Int'l Conf. on Machine Learning*, 2012.

[40] T. Yao, Y. Pan, C.-W. Ngo, H. Li, and T. Mei, "Semi-supervised domain adaptation with subspace learning for visual recognition," in *IEEE Conf. on Computer Vision & Pattern Recognition*, 2015, pp. 2142–2150.

[41] M. Gönen and A. A. Margolin, "Kernelized Bayesian transfer learning," in *Assoc. for the Adv. of Artificial Intelligence*, 2014.

[42] M. Kandemir, "Asymmetric transfer learning with deep Gaussian processes," in *Int'l Conf. on Machine Learning*, 2015.

[43] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popović, "Style-based inverse kinematics," vol. 23, no. 3, pp. 522–531, 2004.

[44] R. Urtasun, D. J. Fleet, and P. Fua, "3D people tracking with gaussian process dynamical models," in *IEEE Conf. on Computer Vision & Pattern Recognition*, 2006, pp. 238–245.

[45] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "MultiPIE," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.

[46] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn, "DISFA: A spontaneous facial action intensity database," *IEEE Trans. on Affective Computing*, vol. 4, no. 2, pp. 151–160, 2013.

[47] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu, and J. M. Girard, "BP4D-spontaneous: a high-resolution spontaneous 3D dynamic facial expression database," *Image and Vision Computing*, vol. 32, no. 10, pp. 692–706, 2014.

[48] M. F. Valstar, T. Almaev, J. M. Girard, G. McKeown, M. Mehu, L. Yin, M. Pantic, and J. F. Cohn, "FERA 2015 - second facial expression recognition and analysis challenge," in *IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, vol. 6, 2015, pp. 1–8.

[49] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "A semi-automatic methodology for facial landmark annotation," in *IEEE Conf. on Computer Vision & Pattern Recognition, Workshops*, 2013.

[50] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,"

IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971–987, 2002.

Stefanos Eleftheriadis received his PhD from the Department of Computing, Imperial College London, UK in 2017 and his Diploma in Electrical and Computer Engineering from Aristotle University of Thessaloniki, Greece, in 2011. He received for his work the national award in Microsoft's Imagine Cup software development competition, in 2011. His current research interests are in machine learning with applications to autonomous decision making and automatic human behavior analysis.

Ognjen (Oggi) Rudovic received his Ph.D. in Computing from Imperial College London, U.K., in 2014. He is currently a Marie Curie Postdoctoral Fellow at MIT Media Lab, working in Affective Computing Group. His research interests are in machine learning and computer vision, and their applications to human-robot interaction, health-care and personalized learning.

Marc Peter Deisenroth is a Lecturer (Assistant Professor) in Statistical Machine Learning at the Department of Computing at Imperial College London. He has been awarded an Imperial College Research Fellowship in 2014 and received Best Paper Awards at ICRA 2014 and ICCAS 2016. He is a recipient of a Google Faculty Research Award and a Microsoft Ph.D. Scholarship.

Maja Pantic is a professor in affective and behavioral computing in the Department of Computing at Imperial College London, UK, and in the Department of Computer Science at the University of Twente, the Netherlands. She currently serves as the editor in chief of Image and Vision Computing Journal and as an associate editor for both the IEEE Transactions on Pattern Analysis and Machine Intelligence and the IEEE Transactions on Affective Computing. She has received various awards for her work on automatic analysis of human behavior, including the European Research Council Starting Grant Fellowship 2008 and the Roger Needham Award 2011. She is a Fellow of the IEEE.