

# Discriminative Shared Gaussian Processes for Multi-view and View-invariant Facial Expression Recognition

Stefanos Eleftheriadis, *Student Member, IEEE*, Ognjen Rudovic, *Member, IEEE*, and Maja Pantic, *Fellow, IEEE*

**Abstract**—Images of facial expressions are often captured from various views as a result of either head movements or variable camera position. Existing methods for multi-view and/or view-invariant facial expression recognition typically perform classification of the observed expression by using either classifiers learned *separately* for each view or a single classifier learned for all views. However, these approaches ignore the fact that different views of a facial expression are just different manifestations of the same facial expression. By accounting for this redundancy, we can design more effective classifiers for the target task. To this end, we propose a Discriminative Shared Gaussian Process Latent Variable Model (DS-GPLVM) for multi-view and view-invariant classification of facial expressions from multiple views. In this model, we first learn a discriminative manifold shared by multiple views of a facial expression. Subsequently, we perform facial expression classification in the expression manifold. Finally, classification of an observed facial expression is carried out either in the view-invariant manner (using only a single view of the expression) or in the multi-view manner (using multiple views of the expression). The proposed model can also be used to perform fusion of different facial features in a principled manner. We validate the proposed DS-GPLVM on both posed and spontaneously displayed facial expressions from three publicly available datasets (MultiPIE, LFPW, and SFEW). We show that this model outperforms the state-of-the-art methods for multi-view and view-invariant facial expression classification, and several state-of-the-art methods for multi-view learning and feature fusion.

**Index Terms**—view-invariant, multi-view learning, facial expression recognition, Gaussian Processes.

## I. INTRODUCTION

FAcial expression recognition (FER) has attracted significant research attention because of its usefulness in many applications, such as human-computer interaction, security and analysis of social interactions, among others [1], [2]. Most existing methods deal with imagery in which the depicted persons are relatively still and exhibit posed expressions in a nearly frontal pose [3]. However, many real-world applications relate to spontaneous interactions (*e.g.*, meeting summarization, political debates analysis, etc.), in which people tend to move their head while being recorded. Furthermore, depending on the camera position, facial images can be taken from multiple views. For these reasons, there is an ever growing

need for automated systems that can accurately perform multi-view and view-invariant facial expression recognition.

The main challenge here is to perform decoupling of the rigid facial changes due to the head-pose and non-rigid facial changes due to the expression, as they are non-linearly coupled in 2D images [4]. Another challenge is how to effectively exploit the information from multiple views (or different facial features) in order to facilitate the expression classification. Thus, accounting for the fact that each view of a facial expression is just a different manifestation of the same underlying facial expression related content is expected to result in more effective classifiers for the target task.

To date, only a few works that deal with multi-view and/or view-invariant FER have been proposed. These focus mainly on recognition of facial expressions of the six basic emotions [5]. Based on how they deal with variation in head-pose (view) and expressions in 2D images, they can be divided into: (i) methods that perform view-invariant, *i.e.*, *per-view*, FER ([6], [7], [8]), (ii) methods that perform the view normalization before performing FER ([9], [10]), and (iii) methods that learn a single classifier using data from multiple views ([11], [12]). However, the main downside of these approaches is that they fail to explicitly model relationships between different views. This, in turn, results in classifiers that are less robust for the target task, but also more complex in the case of large number of views/expressions. All this can efficiently be ameliorated using the modeling strategy of multi-view learning methods (*e.g.*, [13], [14]).

In this work, we introduce the Discriminative Shared Gaussian Process Latent Variable Model (DS-GPLVM) for multi-view and view-invariant FER. We adopt the multi-view learning strategy in order to represent multi-view facial expression data on a common expression manifold. To this end, we use the notion of Shared GPs [15], [16], the generative framework for discovering a non-linear subspace shared across different observation spaces (*e.g.*, the facial views or feature representations). Since our ultimate goal is the expression classification, we place a discriminative prior, informed by the expression labels, over the manifold. The classification of an observed expression is then performed in the learned manifold using the  $k$ NN classifier. The proposed model is a generalization of the discriminative GP Latent Variable Models (D-GPLVM) [17] for non-linear dimensionality reduction and classification of data from a single observation space. The learning of DS-GPLVM is carried out using the expression data from multiple

S. Eleftheriadis, O. Rudovic and M. Pantic are with the Department of Computing, Imperial College London, 180 Queen's Gate, London SW7 2AZ, UK. E-mail: {s.eleftheriadis, o.rudovic, m.pantic}@imperial.ac.uk.

M. Pantic is also with the Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, The Netherlands

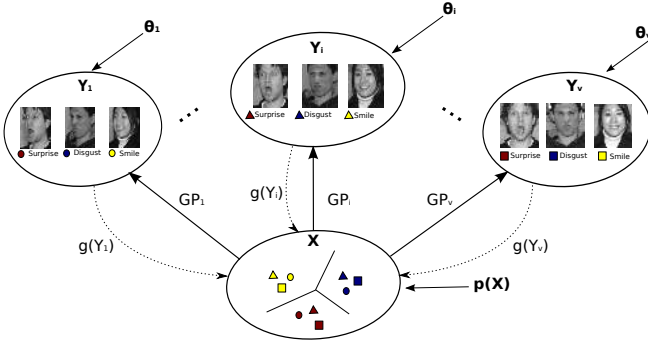


Fig. 1. The overview of the proposed DS-GPLVM. The discriminative shared manifold  $\mathbf{X}$  of facial expressions captured at different views ( $\mathbf{Y}_i$ ,  $i = 1 \dots V$ ) is learned using the framework of shared GPs ( $\text{GP}_i$ ). The class separation in the shared manifold is enforced by the discriminative shared prior  $p(\mathbf{X})$ , informed by the data labels. During inference, the facial images from different views are projected onto the shared manifold by using the kernel-based regression, learned for each view separately ( $g(\mathbf{Y}_i)$ ) for view-invariant approach, or simultaneously from multiple views for multi-view approach. The classification of the query image is then performed using the  $k$ NN classifier.

views. Classification of an observed facial expression, however, can be carried out either in the view-invariant manner (in case only a single view of the observed expression is available at runtime) or in the multi-view manner (in case multiple views of the observed expression are available at runtime). The proposed model can also perform fusion of different facial features in order to improve view-invariant facial expression classification. In order to keep the model computationally tractable in the presence of large number of views, we propose a learning algorithm that splits the learning into different sub-problems (for each view), and then employs the Alternating Direction Method (ADM) [18] to optimize each sub-problem separately. The outline of the proposed approach is given in Fig. 1.

The contributions of this work can be summarized as follows.

- 1) We propose the DS-GPLVM for multi-view and/or view-invariant FER. The proposed model is a generalization of existing discriminative dimensionality reduction methods from single to multiple observation spaces. This is, also, the first approach that exploits the multi-view learning strategy in the context of multi-view FER.
- 2) We propose a novel learning algorithm for efficient optimization of the model parameters that is based on the ADM strategy. This allows us to solve the model parameters' optimization problem for each-view, as a separate sub-problem, to perform parameter optimization for each view separately, resulting in the model being computationally efficient even in the case of a large number of views.
- 3) The proposed DS-GPLVM is applicable to a variety of tasks (multi-view classification, multiple-feature fusion, pose-wise classification, etc.). Compared to state-of-the-art methods for multi-view learning, which employ linear techniques to align different views on a manifold, the DS-GPLVM is a kernel-based method, being able to discover non-linear correlations between different views.

In contrast to state-of-the-art methods for view-invariant and/or multi-view FER, the DS-GPLVM exploits dependencies between different views, improving the FER performance.

Note that an earlier version of this work appeared in [19]. There are two major extensions introduced: 1) in [19], the projections of data from different views to the shared space are learned independently of the manifold, while in the DS-GPLVM proposed here they are learned simultaneously. We show in our experiments that this results in improved recognition of the target facial expressions. 2) Our previous work in [19] is capable only of view-invariant FER, while here we generalize it to the multi-view and feature fusion settings.

Finally, we use the GPs as a basis for our (non-parametric) multi-view learning framework because, in contrast to majority of parametric models, it allows us to capture subtle details of facial expressions and preserve them on the expression manifold that is largely robust to the view/subject differences. Furthermore, due to the probabilistic nature of GPs, different types of priors can seamlessly be integrated into the model for multi-view learning (in our case, discriminative priors over the expression manifold). Last but not least, GPs are known for their ability to generalize quite well even from a small number of training data (on the order of several hundreds) [17]. While this may not seem a big advantage when data are abundant, it is of crucial importance for multi-view FER due to the scarcity of existing datasets containing annotated expressions and poses.

The remainder of the paper is organized as follows. Section II gives an overview of the related work. In Section III we present the theoretical background of the base GPLVM and the D-GPLVM. In Section IV, we introduce the proposed Discriminative Shared Gaussian Process Latent Variable Model for multi-view FER. Section V describes the conducted experiments and shows the results obtained. Finally, in Section VI we conclude the paper.

## II. RELATED WORK

### A. Multi-view and View-invariant FER

As mentioned above, recent advances toward multi-view facial expression recognition can be divided into three groups. A representative of the first group is [6], where the authors used Local Binary Patterns (LBP) [20] (and its variants) to perform a two-step facial expression classification. In the first step, they select the closest head-pose to the (discrete) training pose/view by using the Support Vectors Machine (SVM) [21] classifier. Once the view is known, they apply the view-specific SVM to perform facial-expression classification. In [7], different appearance features, *e.g.*, Scale Invariant Feature Transform (SIFT) [22], Histogram of Oriented Gradients (HOG) [23], LBP, are extracted around the locations of characteristic facial points, and used to train various pose-specific classifiers. Similarly, [8] used per-view-trained 2D Active Appearance Models (AAMs) [24] to locate a set of characteristic facial points, and extract LBP, SIFT and Discrete Cosine Transform (DCT) [25] features around them. By learning separate classifiers for each view, these approaches ignore correlations across different views, which makes them suboptimal for the target task. As

shown by [6], [7], classification of some facial expressions can be performed better in 15° view than in the frontal view, for instance. Hence, the data from more discriminative views for expression classification can be used during learning to improve the underperforming expression classification in the other views. In the proposed DS-GPLVM, we do so by performing the classification in a discriminative feature space shared across views.

The approaches in the second group ([9], [10]) first perform view normalization, and then apply facial expression classification in the canonical view, usually chosen to be the frontal. For the view normalization, the authors propose the Coupled GP (CGP) regression model that exploits pairwise correlations between the views in order to learn robust mappings for projecting facial features (*i.e.*, a set of facial points) from non-frontal to the frontal view. A limitation of this approach is that the view normalization and learning of the expression classifier are done independently, thus bounding the accuracy of the expression classification by that of the view normalization. Also, since the view normalization is performed directly in the observed space, errors in the view normalization step can adversely affect the classification. This is even more so due to the high-dimensional noise affecting the view normalized features. Furthermore, the canonical view has to be selected in advance. This can further limit the accuracy of the expression classification as such view may not be the most discriminative for classification of certain facial expression categories, as mentioned above. These limitations are addressed by the proposed DS-GPLVM, which avoids the need for a canonical view as it performs the classification on a shared manifold of facial expressions from multiple views, the topology of which is optimized for classification of the target expressions.

In the third group of methods ([11], [12]), a single classifier is learned using the expression data from multiple views. Specifically, [11] used variants of dense SIFT [26] features extracted from multi-view facial expression images. Likewise, [12] used the Generic Sparse Coding scheme ([27]) to learn a dictionary that sparsely encodes the SIFT features extracted from facial images in different views. However, because of high variation in appearance of facial expressions in different views and of different subjects, the complexity of the learned classifier increases significantly with the number of views/expressions. This can easily lead to overfitting, and, in turn, poor generalization of the classifier to unseen data. On the other hand, the complexity of the classifier in DS-GPLVM is reduced by accounting for underlying structure of the data (*e.g.*, the correspondences between the views) via the shared manifold.

## B. Multi-view Learning

In what follows, we make a short overview of the most popular multi-view learning methods that can be applied to the multi-view FER. A common approach in multi-view classification is to learn the view-specific projection using paired samples from different views, and to project those samples onto a common latent space, followed by their classification.

The paired samples usually refer to samples that come from the same subject (*e.g.*, face images of a person in two different views). The goal here is to learn a latent space where the paired samples are placed close if they come from the same class/subject, and far apart otherwise.

A widely used unsupervised approach to learn such latent spaces is Canonical Correlation Analysis (CCA) [28] and its non-linear variant Kernel CCA (KCCA) [29]. The goal of these methods is to find projection to a common subspace where the correlation between the low-dimensional embeddings is maximized. These methods can handle data only in the pair-wise manner (thus, only two views at the time), which makes them unfit for multi-view classification problems with more than two views. A generalization of CCA to the multi-view setting, Multiview CCA (MCCA), has been proposed in [30]. The main idea of MCCA is to find a common subspace where the correlation between the low-dimensional embeddings of any two views is maximized. Apart from CCA-based methods, there are a few works that extend the single-view subspace learning to the multi-view case. [31] is a representative of this approach. It is a spectral clustering approach for the multi-view setting. In particular, the spectral embedding from one view is used to constrain the data of the other view. Note that the methods mentioned above are proposed for unsupervised learning. Thus, in the context of the multi-view FER, they are not expected to perform well as the view alignment by these methods is not optimized for classification.

Another group of methods performs supervised multi-view analysis. For instance, Multi-view Fisher Discriminant Analysis (MFDA) [32] learns classifiers in different views, by maximizing the agreement between the predicted labels of these classifiers. However, MFDA can only be used for binary problems. In [14], the authors extended Linear Discriminant Analysis (LDA) [33] to the multiview case, named Multi-view Discriminant Analysis (MvDA). This model maximizes the between-class and minimizes the within-class variations, across all the views, in the common subspace. Generalized Multiview Analysis (GMA) [13] has also been proposed for extending dimensionality reduction techniques for single views to multiple views. An instance of GMA, the Generalized Multiview LDA (GMLDA), finds a set of projections in each view that attempt to separate the content of different classes and unite different views of the same class in a common subspace. Another example of GMA is the GM Locality Preserving Projections (GMLPP), that extends the LPP [34] model, which can be used to find a discriminative data manifold using the labels. Although effective in some tasks, these models are all based on linear projection functions. This can limit their performance when dealing with high-dimensional input features (*i.e.*, appearance based facial features), as well as their ability to successfully unravel non-linear manifold(s) of multiple views. All this is addressed by the proposed DS-GPLVM model.

### III. THEORETICAL BACKGROUND: GAUSSIAN PROCESS LATENT VARIABLE MODELS (GPLVM)

In this section, we first give a brief overview of the GPLVM [35] for learning a non-linear low-dimensional manifold of a single observation space (*e.g.*, the facial expression data from a single view). We then describe two types of discriminative priors for the manifold, which are used to obtain the discriminative GPLVMs [17], [36] for data classification.

#### A. GPLVM

The GPLVM [35] is a probabilistic model for non-linear dimensionality reduction. It learns a low dimensional manifold  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathcal{R}^{N \times q}$ , with  $q \ll D$ , corresponding to the high-dimensional observation space  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N]^T \in \mathcal{R}^{N \times D}$ . The learning of the manifold and its mapping to the observation space is modeled using the framework of Gaussian Processes (GP) [37]. Specifically, by using the covariance function  $k(\mathbf{x}_i, \mathbf{x}_j)$  of GPs, the likelihood of the observed data, given the manifold, is defined as

$$p(\mathbf{Y}|\mathbf{X}, \theta) = \frac{1}{\sqrt{(2\pi)^{ND}|\mathbf{K}|^D}} \exp\left(-\frac{1}{2}\text{tr}(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T)\right), \quad (1)$$

where  $\mathbf{K}$  is the kernel matrix, the elements of which are obtained by applying the covariance function  $k(\mathbf{x}_i, \mathbf{x}_j)$ , to each training data-pair  $(i, j) \in \{1 \dots N\}$ . The covariance function is usually chosen as the sum of the Radial Basis Function (RBF) kernel, bias and noise terms

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_1 \exp\left(-\frac{\theta_2}{2}\|\mathbf{x}_i - \mathbf{x}_j\|^2\right) + \theta_3 + \frac{\delta_{i,j}}{\theta_4}, \quad (2)$$

where  $\delta_{i,j}$  is the Kronecker delta function, and  $\theta = (\theta_1, \theta_2, \theta_3, \theta_4)$  are the kernel parameters [37]. The manifold  $\mathbf{X}$  is then obtained as the mean of the posterior distribution

$$p(\mathbf{X}, \theta|\mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{X}, \theta)p(\mathbf{X}) \quad (3)$$

where the spherical Gaussian prior is usually placed over the manifold. This prior prevents the GPLVM from placing latent points infinitely far apart, *i.e.* latent positions close to the origin are preferred [17]. The learning of the manifold is accomplished by minimizing the negative log-likelihood of the posterior in Eq. (3), w.r.t. the latent coordinates in  $\mathbf{X}$ , which is given by

$$L = \frac{D}{2} \ln |\mathbf{K}| + \frac{1}{2} \text{tr}(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T) - \log(p(\mathbf{X})). \quad (4)$$

To enforce the latent positions to be a smooth function of the data space, [38] proposed to back-constrain the GPLVM. This ensures that the points that are close in the data space are also close on the manifold. More importantly, these constraints allow us to learn the inverse mappings, which are used during the inference step to map the query points from the data space onto the manifold. Specifically, each datum  $\mathbf{y}_i$  is back-constrained so that it satisfies

$$x_{ij} = g_j(\mathbf{y}_i; \mathbf{A}_j) = \sum_{m=1}^N a_{mj} k_{bc}(\mathbf{y}_i, \mathbf{y}_m), \quad (5)$$

where  $x_{ij}$  is the  $j$ -th dimension of  $\mathbf{x}_i \in \mathcal{R}^q$ ,  $g_j$  is the kernel based regression over  $\mathbf{Y}$ , and  $\mathbf{A}$  is the matrix that holds the

parameters for the regression. Different projection vectors  $\mathbf{A}_j$  are used for each feature dimension in order to be able to learn different weights for each feature dimension, as in the standard linear kernel regression. To obtain a smooth inverse mapping in the back-constraints, we use the RBF kernel

$$k_{bc}(\mathbf{y}_i, \mathbf{y}_m) = \exp\left(-\frac{\gamma}{2}\|\mathbf{y}_i - \mathbf{y}_m\|^2\right), \quad (6)$$

where  $\gamma$  is the inverse width parameter. With such defined back constraints, the model learning is accomplished either by minimizing the likelihood in Eq.(4) s.t. the back constraints, or by plugging the expression in Eq.(5) into the likelihood function, and solving the unconstrained optimization problem.

#### B. Discriminative GPLVM (D-GPLVM)

The GPLVM is a generative model of the data, where a simple spherical Gaussian prior is placed over the manifold [17]. However, this model can be adapted for classification by using a discriminative prior that encourages the latent positions of the examples of the same class to be close and those of different classes to be far on the manifold. This has firstly been explored in [17], where a prior based on Linear Discriminant Analysis (LDA) is proposed. LDA tries to maximize between-class separability and minimize within-class variability by maximizing

$$J(\mathbf{X}) = \text{tr}(\mathbf{S}_w^{-1}\mathbf{S}_b), \quad (7)$$

where  $\mathbf{S}_w$  and  $\mathbf{S}_b$  are within- and between-class matrices, respectively, defined as

$$\mathbf{S}_w = \sum_{i=1}^L \frac{N_i}{N} \left[ \frac{1}{N_i} \sum_{k=1}^{N_i} (x_k^{(i)} - \mathbf{M}_i)(x_k^{(i)} - \mathbf{M}_i)^T \right], \quad (8)$$

$$\mathbf{S}_b = \sum_{i=1}^L \frac{N_i}{N} (\mathbf{M}_i - \mathbf{M}_0)(\mathbf{M}_i - \mathbf{M}_0)^T. \quad (9)$$

Here,  $N_i$  training points from class  $i$  are stored in  $\mathbf{X}^{(i)} = [x_1^{(i)}, \dots, x_{N_i}^{(i)}]$ ,  $\mathbf{M}_i$  is the mean of examples of class  $i$ , and  $\mathbf{M}_0$  is the mean of examples of all the classes. The energy function in Eq. (7) is used to define discriminative prior over the manifold as

$$p(\mathbf{X}) = \frac{1}{Z_d} \exp\left\{-\frac{1}{\sigma_d^2} J^{-1}\right\}, \quad (10)$$

where  $Z_d$  is a normalization constant, and  $\sigma_d$  represents a global scaling of the prior. Then, the Discriminative GPLVM (D-GPLVM) [17] is obtained by replacing the Gaussian prior in Eq. (3) with the prior in Eq. (10). The authors also proposed a version of the prior based on Generalized Discriminant Analysis (GDA).

A more general prior based on the notion of the graph Laplacian matrix [39] has been used to derive a discriminative GPLVM model named Gaussian Process Latent Random Field (GPLRF) [36]. To define the prior, an undirected graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is first constructed, where  $\mathcal{V} = \{V_1, V_2, \dots, V_N\}$  is the node set, with node  $V_i$  corresponding to a training example  $\mathbf{x}_i$ , and  $\mathcal{E} = \{(V_i, V_j)_{i,j=1 \dots N} | i \neq j, \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ belong to the same class}\}$  is the edge set. By pairing each node with the random vector  $\mathbf{X}_{*k} = (\mathbf{X}_{1k}, \mathbf{X}_{2k}, \dots, \mathbf{X}_{Nk})^T$

(for  $k = 1, 2, \dots, q$ ), we obtain a Gaussian Markov Random Field (GMRF) [40] w.r.t. graph  $\mathcal{G}$ . Next, each edge in the graph is associated with a weight (in this case, 1), and the weights are stored in the weight matrix defined as

$$\mathbf{W}_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j, i \neq j, \text{ belong to the same class} \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

The graph Laplacian matrix is then defined as  $\mathbf{L} = \mathbf{D} - \mathbf{W}$ , where  $\mathbf{D}$  is a diagonal matrix with  $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$ . Finally, using  $\mathbf{L}$ , the discriminative GMRF prior is defined as

$$p(\mathbf{X}) = \prod_{k=1}^q p(\mathbf{X}_{*k}) = \frac{1}{Z_q} \exp \left[ -\frac{\beta}{2} \text{tr}(\mathbf{X}^T \mathbf{L} \mathbf{X}) \right], \quad (12)$$

where  $Z_q$  is a normalization constant and  $\beta > 0$  is a scaling parameter. The term  $\text{tr}(\mathbf{X}^T \mathbf{L} \mathbf{X})$  in the discriminative prior in Eq. (12) reflects the sum of the distances between the latent positions of the examples from the same class. Thus, the latent positions from the same class that are closer will be given higher probability. This prior can be seen as a more general version of the LDA prior in Eq. (10), without the restriction on the size of the manifold. Also, the weights used to compute  $\mathbf{L}$  can be defined using not only the labels, but also the observed data, resulting in additional smoothing constraints. Finally, the cost function of the GPLRF model is obtained by plugging the prior in Eq. (12) into Eq. (4).

#### IV. DISCRIMINATIVE SHARED GPLVM (DS-GPLVM)

The D-GPLVM from Sec. III-B is designed for a single observation space. In this section, we generalize the D-GPLVM so that it can simultaneously learn a discriminative manifold of multiple observation spaces. This is attained by using the framework of Shared GPs ([15], [16]). In our approach, we assume that the multiple observation spaces (*e.g.*, different views of facial expressions) are dependent, and that they can be aligned on a discriminative shared manifold. In what follows, we first introduce the Shared GP model for alignment (fusion) of multiple observation spaces in the shared manifold, and define the discriminative shared-space prior for the manifold. We then describe learning and inference in the proposed model.

##### A. Shared-space GPLVM

Given a set of corresponding features  $\mathbf{Y} = \{\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(V)}\}$ , extracted from  $V$  views, instead of learning independent manifold of data from each view as done in GPLVM, we learn a single manifold  $\mathbf{X}$  that is assumed to be shared among the views. Within the Shared GPs framework, the joint likelihood of  $\mathbf{Y}$ , given the shared manifold  $\mathbf{X}$ , is factorized as follows

$$p(\mathbf{Y}|\mathbf{X}, \theta_s) = p(\mathbf{Y}_1|\mathbf{X}, \theta^{(1)}) \dots p(\mathbf{Y}_V|\mathbf{X}, \theta^{(V)}), \quad (13)$$

where  $\theta_s = \{\theta^{(1)}, \dots, \theta^{(V)}\}$  are the kernel parameters for each observation space, and the kernel function is defined as in Eq. (2). It is assumed here that each observation space is generated from the shared manifold via separate GP. The shared latent space  $\mathbf{X}$  is then found by minimizing the joint

negative log-likelihood penalized with the prior placed over the shared manifold, and is given by

$$L_s = \sum_v L^{(v)} - \log(p(\mathbf{X})) \quad (14)$$

where  $L^{(v)}$  is the negative log-likelihood of data from view  $v = 1, \dots, V$ , and is given by

$$L^{(v)} = \frac{D}{2} \ln |\mathbf{K}^{(v)}| + \frac{1}{2} \text{tr}[(\mathbf{K}^{(v)})^{-1} \mathbf{Y}^{(v)} (\mathbf{Y}^{(v)})^T] + \frac{ND}{2} \ln 2\pi, \quad (15)$$

where  $\mathbf{K}^{(v)}$  is the kernel matrix associated with the input data  $\mathbf{Y}^{(v)}$ . In Eq. (15), the spherical Gaussian prior is placed over the manifold. To obtain a shared manifold for multi-view classification, in the following we define a discriminative shared-space prior.

##### B. Discriminative Shared-space Prior

To define discriminative shared-space prior for multi-view learning, we generalize the GMRF prior for the single view given by Eq. (11). To this end, we first construct the view-specific weight matrices  $\mathbf{W}^{(v)}$ ,  $v = 1, \dots, V$ . Instead of using only the class labels, we also use the data-dependent weights. Specifically, the elements of the weight matrix are obtained by applying the RBF kernel to the data from each view as

$$\mathbf{W}_{ij}^{(v)} = \begin{cases} \exp \left( -\frac{\|\mathbf{y}_i^{(v)} - \mathbf{y}_j^{(v)}\|^2}{t^{(v)}} \right) & \text{if } i \neq j \text{ and } c_i = c_j, \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

where  $\mathbf{y}_i^{(v)}$  is the  $i$ -th sample (row) in  $\mathbf{Y}^{(v)}$ ,  $c_i$  is the class label, and  $t^{(v)}$  is the kernel width which is set to the mean squared distance between the training inputs as in [41]. Then, the graph Laplacian for view  $v$  is  $\mathbf{L}^{(v)} = \mathbf{D}^{(v)} - \mathbf{W}^{(v)}$ , where  $\mathbf{D}^{(v)}$  is a diagonal matrix with  $\mathbf{D}_{ii}^{(v)} = \sum_j \mathbf{W}_{ij}^{(v)}$ . Because the graph Laplacians from different views vary in their scale, we use the normalized graph Laplacian, defined as

$$\mathbf{L}_N^{(v)} = (\mathbf{D}^{(v)})^{-1/2} \mathbf{L}^{(v)} (\mathbf{D}^{(v)})^{-1/2}, \quad (17)$$

Subsequently, we define the (regularized) joint Laplacian as

$$\tilde{\mathbf{L}} = \mathbf{L}_N^{(1)} + \mathbf{L}_N^{(2)} + \dots + \mathbf{L}_N^{(V)} + \xi \mathbf{I} = \sum_v \mathbf{L}_N^{(v)} + \xi \mathbf{I}, \quad (18)$$

with  $\mathbf{I}$  the identity matrix, and  $\xi$  a regularization parameter (typically set to a small value *e.g.*,  $10^{-4}$ ), which ensures that  $\tilde{\mathbf{L}}$  is positive-definite [42]. This, in turn, allows us to define the discriminative shared-space prior as

$$p(\mathbf{X}) = \prod_{v=1}^V p(\mathbf{X}|\mathbf{Y}^{(v)})^{\frac{1}{V}} = \frac{1}{V \cdot Z_q} \exp \left[ -\frac{\beta}{2} \text{tr}(\mathbf{X}^T \tilde{\mathbf{L}} \mathbf{X}) \right]. \quad (19)$$

Here,  $Z_q$  is a normalization constant and  $\beta > 0$  is a scaling parameter. The discriminative shared-space prior in (19) aims at maximizing the class separation in the manifold learned from data from all the views, and it can be regarded as a multi-view kernel extension of the parametric LDA/LPP prior defined for a single view in [17], [36]. Using this prior, the

negative log-likelihood of the proposed DS-GPLVM model is given by

$$L_s(\mathbf{X}) = \sum_v L^{(v)} + \frac{\beta}{2} \text{tr}(\mathbf{X}^T \tilde{\mathbf{L}} \mathbf{X}), \quad (20)$$

where  $L^{(v)}$  is defined by Eq. (15).

### C. Back-constraints

In the GPLVM from Sec.III-A, the back-constraints, defined by the inverse mappings, ensure that topology of the output space is preserved on the manifold. In DS-GPLVM, this is achieved by the discriminative shared-space prior since the weight matrix used to define the prior is built from input data. However, to perform inference with DS-GPLVM we still need to learn the inverse mappings that project data from different views onto the shared manifold. For this, we consider two scenarios. In the first, we define  $v$  sets of constraints (one for each view), which are enforced by separate inverse mappings from each view to the shared space. In the second, we define one set of constraints (for all the views), and which are enforced by a single inverse mapping from all the views to the shared space. We refer to the former as independent back-projections (IBP), and the latter as single back-projection (SBP). These are given by

- **IBP** from each view  $v = 1, \dots, V$

$$\mathbf{X} = g(\mathbf{Y}^{(v)}, \mathbf{A}^{(v)}) = \mathbf{K}_{bc}^{(v)} \mathbf{A}^{(v)}. \quad (21)$$

- **SBP** from  $V$  views

$$\mathbf{X} = g(\mathbf{Y}, \mathbf{A}) = \left( \sum_{v=1}^V w_v \mathbf{K}_{bc}^{(v)} \right) \mathbf{A} = \tilde{\mathbf{K}} \mathbf{A}, \quad (22)$$

where  $g(\cdot, \cdot)$  represents the mapping function(s) learned using the kernel regression. The elements of  $\mathbf{K}_{bc}^{(v)}$  are given by Eq. (6) and  $w_v$  is the (scalar) weight for view  $v$ .

Note that for a single view, the model can be reparametrized to obtain an unconstrained optimization problem (see Sec. III-A). Yet, in the case of multiple views, this is not possible as it would result in different  $\mathbf{X}$  for each view. Therefore, we need to solve a constrained optimization problem, the complexity of which increases with the number of views. To efficiently solve this, in the following section we propose an iterative learning algorithm for simultaneous learning of the shared space and inverse mappings in the proposed model.

### D. DS-GPLVM: Learning and Inference

Learning of the model parameters  $\mathbf{X}, \theta_s$  and  $\mathbf{A}$ , consists of minimizing the negative log-likelihood given by Eq. (20) subject to either the IBP or SBP constraints. Formally, we aim to solve the following minimization problem:

$$\begin{aligned} & \arg \min_{\mathbf{X}, \theta_s, \mathbf{A}} L_s(\mathbf{X}) + R(g) \quad (23) \\ \text{s.t.} \quad & \begin{cases} IBP(\mathbf{X}, \mathbf{A}^{(v)}) \triangleq \mathbf{X} - \mathbf{K}_{bc}^{(v)} \mathbf{A}^{(v)} = \mathbf{0}, v = 1, \dots, V \\ SBP(\mathbf{X}, \mathbf{A}) \triangleq \mathbf{X} - \tilde{\mathbf{K}} \mathbf{A} = \mathbf{0}, \sum_{v=1}^V w_v = 1, w_v \geq 0, \end{cases} \end{aligned}$$

where  $R(g)$  is a regularization term. To obtain the function form for  $R(g)$ , we first derive the solution of the regularized kernel regression from the mapping function of the infinite-dimensional feature space  $g(\mathbf{x}_i) = \phi(\mathbf{x}_i)^T w$ , as in [43]. The solution to this problem is of the form of  $w = \sum_{i=1}^N a_i \phi(\mathbf{x}_i)$ . Hence, by applying the Representer Theorem [44] on this space, and by using the Tikhonov regularization for the parameters  $w$ , we arrive at the optimal functional form for  $R(g)$  as

$$\begin{cases} \sum \frac{\lambda^{(v)}}{2} r(g^{(v)}), & r(g^{(v)}) = \text{tr}((\mathbf{A}^{(v)})^T \mathbf{K}_{bc}^{(v)} \mathbf{A}^{(v)}), \text{ for IBP} \\ \frac{\lambda}{2} \text{tr}(\mathbf{A}^T \tilde{\mathbf{K}} \mathbf{A}) & , \text{ for SBP} \end{cases} \quad (24)$$

*IBP: Parameter Optimization.* We first present the learning procedure for the more general case involving the IBP constraints, and then provide the solution for the SBP case. From Eq. (23), we see that the back-mapping from each view is represented by an independent set of linear constraints. We exploit this to find the model parameters by iteratively solving a set of sub-problems. To this end, we first incorporate the IBP constraints into the regularized log-likelihood in Eq. (23) by using the Lagrange multipliers. As a result, we obtain the following augmented Lagrangian function:

$$\begin{aligned} \mathcal{L}^{IBP}(\mathbf{X}, \{\mathbf{A}^{(v)}, \Lambda^{(v)}\}_{v=1}^V) &= L_s(\mathbf{X}) + R(g) + \\ & \sum_{v=1}^V \langle \Lambda^{(v)}, IBP(\mathbf{X}, \mathbf{A}^{(v)}) \rangle + \frac{\mu}{2} \sum_{v=1}^V \|IBP(\mathbf{X}, \mathbf{A}^{(v)})\|_F^2, \end{aligned} \quad (25)$$

where  $\Lambda^{(v)}$  are the Lagrange multipliers for view  $v$ ,  $\langle \cdot, \cdot \rangle$  is the inner product, and  $\mu > 0$  is the penalty parameter. We can see from Eq. (25) that the linear constraint has been incorporated into the cost function as a quadratic penalty term without affecting the solution to the problem. The role of the Lagrange multipliers (inner product term) is to achieve efficiency in obtaining the solution without the requirement of sequentially increasing the penalty parameter to infinity [18]. The standard approach is to minimize the objective in Eq. (25) w.r.t. all the model's parameters simultaneously. Yet, this is impractical, as the fact that the objective function is separable, is not exploited to simplify the problem. To remedy this, we employ the Alternating Direction Method (ADM) [18] to decompose the minimization into subproblems, each of which can be solved separately w.r.t. to a subset of the model parameters. More specifically, we split the learning of the parameters of the shared space and the back-mappings from each view, by defining the iterations of ADM as follows. We first solve for  $\mathbf{X}$  and  $\theta_s$  as

$$\begin{aligned} \{\mathbf{X}, \theta_s\}_{t+1} &= \arg \min_{\mathbf{X}, \theta_s} L_s(\mathbf{X}) + \\ & \frac{\mu_t}{2} \sum_{v=1}^V \|IBP(\mathbf{X}, \mathbf{A}_t^{(v)}) + \frac{\Lambda_t^{(v)}}{\mu_t}\|_F^2. \end{aligned} \quad (26)$$

Then, for each view  $v = 1, \dots, V$ , we solve for  $\mathbf{A}^{(v)}$  as

$$\mathbf{A}_{t+1}^{(v)} = \arg \min_{\mathbf{A}^{(v)}} r(\mathbf{A}^{(v)}) + \frac{\mu_t}{2} \|IBP(\mathbf{X}_{t+1}, \mathbf{A}^{(v)}) + \frac{\Lambda_t^{(v)}}{\mu_t}\|_F^2, \quad (27)$$

and finally update the Lagrangian and the penalty parameter as

$$\Lambda_{t+1}^{(v)} = \Lambda_t^{(v)} + \mu_t IBP(\mathbf{X}_{t+1}, \mathbf{A}_{t+1}^{(v)}) \quad (28)$$

$$\mu_{t+1} = \min(\mu_{max}, \rho\mu_t), \quad (29)$$

respectively. Note that in Eq. (29),  $\rho$  is kept constant (it is typically set to  $\rho = 1.1$ ).

Since there is not a closed-form solution for the problem in Eq. (26), we use the conjugate gradient algorithm (CG) [37] to minimize the objective w.r.t. the latent positions  $\mathbf{X}$  and the kernel parameters  $\theta_s^1$ . On the other hand, the problem in Eq. (27) is similar to that of Kernel Ridge Regression (KRR), and it has a closed-form solution, which is given by

$$\mathbf{A}^{(v)} = (\mathbf{K}_{bc}^{(v)} + \frac{\lambda^{(v)}}{\mu_t} \mathbf{I})^{-1} (\mathbf{X} + \frac{\Lambda_t^{(v)}}{\mu_t}) \quad (30)$$

However, this solution depends on the parameters  $\gamma^{(v)}$  and  $\lambda^{(v)}$ , which need to be tuned through costly cross-validation procedures. To alleviate this, we reformulate the optimization problem in Eq. (27). For this, we use the notion of the Leave-One-Out (LOO) cross-validation procedure for the KRR [45] to define the learning of the parameters  $\gamma^{(v)}$  and  $\lambda^{(v)}$ . Once estimated, these parameters are used to compute  $\mathbf{A}^{(v)}$ .

The idea of the LOO learning procedure is based on the fact that given any training set and the corresponding learned regression model, if we add a sample to the training set with the target equal to the output predicted by the model, the latter will not change since the cost function will not increase [45]. Thus, given the training set with the sample  $\mathbf{y}_i^{(v)}$  left out, the predicted outputs  $\hat{\mathbf{X}}^{(-i)}$  (the superscript denotes that the  $i$ -th sample was left out) will not change if the sample  $\mathbf{y}_i^{(v)}$  with target  $\hat{\mathbf{x}}_i^{(-i)}$  is added to the set. Then, the goal of LOO is to minimize the difference between the predictions  $\hat{\mathbf{x}}_i^{(-i)}$  and the actual outputs  $\mathbf{x}_i$  for all the samples. To compute this, we first need to define the matrix

$$\mathbf{M} \triangleq \begin{bmatrix} m_{ii} & \mathbf{m}_i^T \\ \mathbf{m}_i & \mathbf{M}_i \end{bmatrix} = (\mathbf{K}_{bc}^{(v)} + \frac{\lambda^{(v)}}{\mu_t} \mathbf{I}), \quad (31)$$

where we partitioned the inverse matrix from Eq. (36) so that the elements corresponding to the  $i$ -th sample appear only in the first row and column of  $\mathbf{M}$  (the same is done for  $\mathbf{X}$  and  $\Lambda_t^{(v)}$  in order to place the  $i$ -th row on the top). Furthermore,  $\mathbf{M}_i$  is the kernel matrix formed from the remaining elements as  $\mathbf{M}_i = (\mathbf{K}_{bc \setminus i}^{(v)} + \frac{\lambda^{(v)}}{\mu_t} \mathbf{I}_{N-1})$ . Then, using Eq. (36), the prediction and the actual target for sample  $i$  are given by

$$\hat{\mathbf{x}}_i^{(-i)} = \mathbf{m}_i^T \mathbf{M}_i^{-1} \mathbf{m}_i \mathbf{A}_i^{(v)} + \mathbf{m}_i^T \mathbf{A}_{-i}^{(v)} \quad (32)$$

$$\mathbf{x}_i = m_{ii} \mathbf{A}_i^{(v)} + \mathbf{m}_i^T \mathbf{A}_{-i}^{(v)} - \Lambda_i^{(v)} / \mu_t. \quad (33)$$

We can now define the cost for the LOO procedure, which is

$$E_{LOO} = \frac{1}{2} \sum_{i=1}^N \|\mathbf{x}_i - \hat{\mathbf{x}}_i^{(-i)}\|^2 = \frac{1}{2} \sum_{i=1}^N \left\| \frac{\mathbf{A}_i^{(v)}}{[\mathbf{M}^{-1}]_{ii}} - \frac{\Lambda_i^{(v)}}{\mu_t} \right\|^2 \quad (34)$$

<sup>1</sup>The derivatives of the objective w.r.t. the model parameters are given in the appendix

---

### Algorithm 1 DS-GPLVM: Learning and Inference

---

#### Learning

Inputs:  $\mathcal{D} = (\mathbf{Y}^{(v)}, \mathbf{c}), v = 1, \dots, V$

Initialize  $\mu_{max} \gg \mu_0 > 0, \rho = const., \mathbf{X}_0, \mathbf{A}_0^{(v)}, \Lambda_0^{(v)}$ .

#### repeat

**Step 1:** Update  $(\mathbf{X}, \theta_s)$  by minimizing Eq. (26).

**Step 2:** Minimize  $E_{LOO}$  from Eq. (34) w.r.t

$(\gamma^{(v)}, \lambda^{(v)})_{v=1, \dots, V}$  for IBP, and  $(\gamma, \lambda)$  for SBP.

**Step 3:** Update  $(\Lambda^{(v)}, \mu, \mathbf{A}^{(v)})$  for IBP, and  $(\Lambda, \mu, \mathbf{A})$  for SBP, from Eq. (28), (29) and (36).

**until** convergence of Eq. (25)

Outputs:  $\mathbf{X}, \mathbf{A}$

---

#### Inference

Inputs:  $\mathbf{y}^{(v)*}$  for IBP, and  $[\mathbf{y}^{(1)*}, \dots, \mathbf{y}^{(V)*}]$  for SBP,  $k$  for classification.

**Step 1:** Find the projection  $\mathbf{x}^*$  to the latent space using Eq. (21) for IBP, and Eq. (22) for SBP.

**Step 2:** Apply kNN classifier to the latent space to obtain the class prediction:  $c^* = \text{kNN}(\mathbf{x}^*, \mathbf{X})$ .

Output:  $c^*$

---

Minimization of  $E_{LOO}$  w.r.t.  $\gamma^{(v)}$  and  $\lambda^{(v)}$  is accomplished using the CG algorithm again.<sup>2</sup> By plugging these parameters into Eq. (36), we obtain  $\mathbf{A}^{(v)}$ . Note that by adopting the LOO learning approach, we: (i) avoid the burden of the standard cross-validation procedures, which are time-consuming, and (ii) reduce the chances of overfitting the model parameters by using the additional cost defined in Eq. (34).

At this point, it is important to clarify that under the proposed ADM-based optimization scheme we are able to automatically learn the majority of the model's parameters (*i.e.*,  $\mathbf{X}, \theta, \mu, \lambda, \gamma$ ), avoiding the need of their tuning via validation procedures. The only parameter learned by means of cross-validation is the weight of the prior,  $\beta$ , while we also need to explore the effect of the dimensionality,  $q$ , of the manifold.

*SBP: Parameter Optimization.* Analogous to the IBP case, we define the Augmented Lagrangian function for the SBP case using the regularized negative log-likelihood and the SBP constraints from Eq. (23). The resulting function has the form as in Eq. (25), but after dropping the dependencies on  $v$ , and replacing the IBP by SBP constraints. The model parameters are then found by applying the proposed ADM to the Augmented Lagrangian function. For this, the objectives in each iteration of the ADM for the IBP case described above are adjusted accordingly.

To achieve efficiency, when applying the CG algorithm to the objective in each iteration of the ADM, with either IBP or SBP constraints, we stop at the first line search of CG, update the corresponding parameters, and go to the next iteration. The ADM cycle is repeated until convergence of the Augmented Lagrangian function.

Inference in the DS-GPLVM is straightforward. The test

<sup>2</sup>The exact derivation of Eq. (32)-(33) along with the gradients of Eq. (34) w.r.t.  $\gamma^{(v)}$  and  $\lambda^{(v)}$  are given in the appendix.

data  $\mathbf{y}^*$  (which for the view-invariant case come from a single view  $v$ , and for the multi-view case from all available views) are first projected to the shared space using the back-mappings defined by Eq. (21) for the IBP, or Eq. (22) for the SBP case. In the second step, classification of the target facial expression is accomplished by using a single classifier trained on the discriminative shared manifold. For this, we use the  $k$ NN classifier<sup>3</sup>. Alg.1 summarizes the learning and inference of the proposed DS-GPLVM.

## V. EXPERIMENTS

### A. Datasets and Experimental Procedure

We evaluate the performance of the proposed DS-GPLVM on expressive face images from three publicly available datasets: MultiPIE [46], Labeled Face Parts in the Wild (LFPW) [47] and Static Facial Expressions in the Wild (SFEW) [48]. Fig. 2 shows sample images from these datasets. From the MultiPIE dataset we used images of 270 subjects depicting acted facial expressions of Neutral (NE), Disgust (DI), Surprise (SU), Smile (SM), Scream (SC) and Squint (SQ), captured at pan angles  $-30^\circ$ ,  $-15^\circ$ ,  $0^\circ$ ,  $15^\circ$  and  $30^\circ$ , resulting in 1531 images per pose. For all images, we selected the flash from the view of the corresponding camera in order to have the same illumination conditions. The LFPW dataset contains images downloaded from google.com, flickr.com, and yahoo.com, depicting spontaneous facial expressions (mainly smiles), in large variation of poses, illumination and occlusion. We used 200 images of NE and SM expressions from the test set provided by [47]. We manually annotated the images in terms of the poses used in MultiPIE. Lastly, the SFEW dataset consists of 700 images of 95 subjects, extracted from movies containing facial expressions with various head poses, occlusions and illumination conditions. The images have been labeled in terms of six basic emotion expressions, *i.e.*, Anger (AN), Disgust (DI), Fear (FE), Happiness (HA), Sadness (SA), Surprise (SU) and Neutral (NE).

The images from both MultiPIE and LFPW were cropped so as to have equal size ( $140 \times 150$  pixels), and annotations of the locations of 68 facial landmark points were provided by [49], which were used to align the facial images in each pose using an affine transform. Similarly, the images from SFEW were cropped ( $112 \times 164$  pixels) and aligned using 5 facial landmark points (center of the eyes, tip of the nose, and corners of the mouth) provided by [48]. For the experiments on MultiPIE, we used three sets of features: (I) facial points, (II) LBPs [20], and (III) DCT [25]. More specifically, from each aligned facial image we extracted LBPs and DCT features from local patches

<sup>3</sup>In the model as defined, the resulting posterior is the manifold and not the class information, so it cannot be used for the classification. For this reason, we need to apply a classifier to the inputs projected onto this manifold during inference. A reasonable choice would be to opt for the GP classifier, however, in our case this would be impractical for two reasons: (i) in the case of more than two classes, the computation complexity of GPC increases significantly since we have to learn a different kernel for each class, making it less applicable to the large number of classes/views. (ii) More importantly, since we are not interested in the classification uncertainty, the GPC is expected to perform similarly to the standard kernel regression, as noted in [37]. Thus, we opt for the deterministic  $k$ NN classifier which is the commonly employed classifier in the GPLVM discriminative models (*e.g.*, see GPLRF [36]).

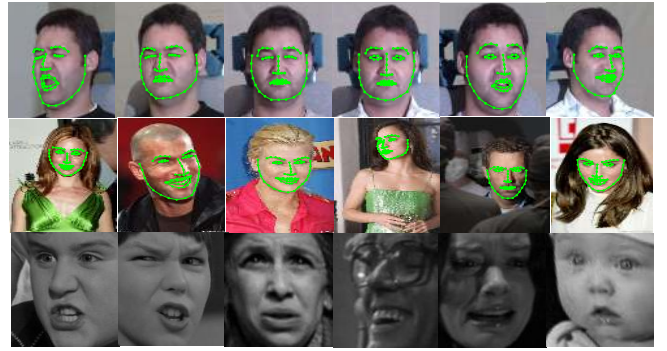


Fig. 2. Example images from MultiPIE (top), LFPW (middle) and SFEW (bottom) datasets with the facial point annotations for the first two.

of size  $15 \times 15$  around the facial landmarks. For LBPs, we used 8 neighbors with radius 2, and in the case of DCT we kept the first 15 coefficients (zig-zag method) of each patch. We then concatenated the results from all the patches, to form the feature vectors. Note that LBP and DCT are complementary features, since the former captures local information between neighborhood of pixels, while the latter preserves the spatial correlation of the pixels inside the neighborhood. Finally, we applied PCA on the three feature sets, keeping 95% of the total energy, to remove unwanted noise and artifacts and reduce the dimensionality of the original feature vectors (especially the appearance based). The resulting dimensionality of each set varies among the views. The point features have around 20 dimensions, while both the appearance features have around 100 dimensions. In the experiments conducted on LFPW, we used only feature set (I), while for SFEW we extracted the same local texture descriptors as in [48], *i.e.*, Local Phase Quantization (LPQ) [50] and Pyramid of HOG (PHOG) [51]. To reduce the dimensionality, we applied again PCA by keeping the same amount of energy, *i.e.*, 95%. This results in 47- and 220-dimensional feature vectors respectively.

The conducted experiments are organized as follows. In Sec.V-B, we perform a qualitative analysis of the DS-GPLVM using the MultiPIE dataset. In Sec.V-C, we evaluate the effectiveness of the proposed DS-GPLVM in the task of multi-view FER on MultiPIE. Specifically, we consider two settings: the standard *multi-view* setting, where images from all the views are available during training/inference, and *view-invariant* setting, where images from all the views are available during training but only a single view is available during inference. Furthermore, we also evaluate the model on the feature fusion task, where different types of features extracted within the same view are used. In addition, we challenge the robustness of the model under different illumination, where we evaluate the performance of the model on images with different lighting conditions within the same view. In Sec.V-E, we test the ability of the DS-GPLVM to generalize to spontaneously displayed facial expressions. For this, we perform the cross-dataset evaluation of the model, where images of SM and NE class from MultiPIE are used for training, and images of the corresponding classes from LFPW for testing. Finally, in Sec.V-F, we evaluate DS-GPLVM on the feature fusion task



using real-world images from the SFEW dataset.

In the experiments mentioned above, we compare the DS-GPLVM to the state-of-the-art view-invariant and multi-view learning methods. As the baseline method, we use the 1-nearest neighbor (1-NN) classifier trained/tested in the original feature space. Similarly, we apply 1-NN classifier to the subspace obtained by LDA [33], supervised LPP [52], and their kernel counterparts, the D-GPLVM [17] with the LDA-based prior, and the GPLRF [36]. These are well-known methods for supervised dimensionality reduction, and we show their performance in the view-invariant version of the experiments. We also compare to our previous work in [19], where the latent space and the back-mappings are learned independently. We denote this model as DS-GPLVM (ind.) to distinguish it from the model proposed here. In the experiments conducted in the multi-view/feature fusion settings, we compare DS-GPLVM to the baseline methods: CCA [28] and KCCA [29]. Since they are designed to deal with only two modalities (feature sets), we follow the pair-wise (PW) evaluation approach, as in [14], *i.e.*, the methods were trained on all combinations of view pairs, and their results were averaged. We also compared DS-GPLVM to the state-of-the-art methods for multi-view learning, namely, the MvDA [14], and the multi-view extensions of LDA (GMLDA), and LPP (GMLPP), proposed in [13].

In all our experiments we performed 5-fold subject independent cross-validation. We used a separate validation set to tune the parameters of each model. More specifically, for all the GPLVM-based methods (*i.e.*, DS-GPLVM, GPLRF and D-GPLVM) the optimal weight for the prior  $\beta$  was set using a grid search. For the GPLRF and D-GPLVM we performed additionally an extra grid search to tune the parameter of the kernel of the back-mapping (RBF kernel was used) as in [17]. For the GMA-based methods (*i.e.*, GMLDA and GMLPP) we tuned the parameter that controls the alignment of the subspaces as suggested in [13]. Finally, in KCCA the width of the employed RBF kernel was cross-validated, while LPP, LDA and MvDA had no parameters to tune. To report the accuracy of FER, we use the classification rate, where the classification was performed on the test set using the 1-NN classifier in all the subspace-based models.

The five folds with the corresponding train, validation and test sets have been generated once and kept fixed during all the experiments for all the methods, in order to achieve a fair comparison. For the experiments on MultiPIE the size of the train, validation and test set was 600, 600, and 300 images per view respectively. For the cross dataset experiment, since we used only images with SM and NE expressions from MultiPIE to train the models, the resulting train and validation sets were slightly smaller, and in particular, 500 and 100 images per pose respectively. The test set was the 200 images from LFPW and it varied depending the pose from 30 – 65 images. Finally, for the experiments on SFEW we adopted the configuration proposed by the creators of the dataset in [48]. The data were already split into two folds, for training and testing. Each time the training fold was further split in 5 folds, to tune the parameters of the models with 5-fold subject independent cross-validation. The size of the resulting sets was 280, 70 and 350 images respectively. For this experiment, due to the small

size of the dataset, after tuning the parameters with the cross validation, each model was re-trained on the whole train and validation set (the one of the two original folds of the dataset) with the optimal parameters, before reporting the results on the test set.

### B. DS-GPLVM: Qualitative Analysis

In this section, we evaluate the performance of the proposed DS-GPLVM w.r.t. the various parameter values. For this, we use the feature set (I), *i.e.*, the facial points, extracted from the MultiPIE dataset. Fig. 3 shows average classification rate (across the views) of the DS-GPLVM for different number of training samples per view, the size of the shared-space, and parameter  $\beta = \{1, 3, 10, 30, 100, 300, 1000, 10000\}$ . Fig. 3(a) shows performance of SBP and IBP versions of DS-GPLVM, the parameters of which are learned using a varying number of training data, while the manifold size is fixed to 5. We see that the SBP versions of DS-GPLVM (*multi-view* setting) achieves a high classification rate ( $\sim 87\%$ ) when using a relatively small number of training data (*i.e.*, 100 images per view). On the other hand, the IBP version of DS-GPLVM (*view-invariant* setting) requires more training data ( $\sim 500$  images per view) to achieve a similar performance. This is a consequence of not using the images from all available views during the inference step. However, with the increased number of training data, the model effectively learns the correlations among the views, rendering the information from some views redundant during the inference. From Fig. 3(b), we see how the size of the shared space affects the accuracy of the learned model. It is clear that both SBP and IBP variants of the model find the 5-dimensional shared-space optimal for classification. Lower dimensional manifolds fail to explain the correlations among the views, while manifolds with more than 5 dimensions do not include any additional discriminative information. Fig. 3(c) illustrates the influence of the shared-space discriminative prior on the classification task. In the case of both SBP and IBP,  $\beta = 300$  results in the best performance of the model, while its further increase leads to a drop in the performance. This is expected, as for high values of  $\beta$  the likelihood term in the DS-GPLVM is fully ignored, resembling LPP. Evidently, such model is prone to overfitting mainly because of the strong influence of the labels during training. On the other hand, for small values of  $\beta$  the shared-space is not sufficiently informed about the class labels, resulting again in a lower performance. In what follows, we set for both the SBP and IBP variants of the model the number of training examples to 500, size of the shared space to 5, and  $\beta = 300$ .

Fig. 3(d)-(f) illustrate the convergence properties of the DS-GPLVM. We see from Fig. 3(d) that the regularized negative log-likelihood of the model reaches a local minimum in less than 25 cycles of the ADM. Fig. 3(e) shows the Frobenius norm [33] of the constraints for the SBP and IBP variants, the difference between the estimated shared space and the back-mappings. Note that the DS-GPLVM is always initialized in the  $-15^\circ$  view (it is found to be the most informative view). Hence, we can see that the norm of this view (black curve) starts from a low value when IBP is used. However, with

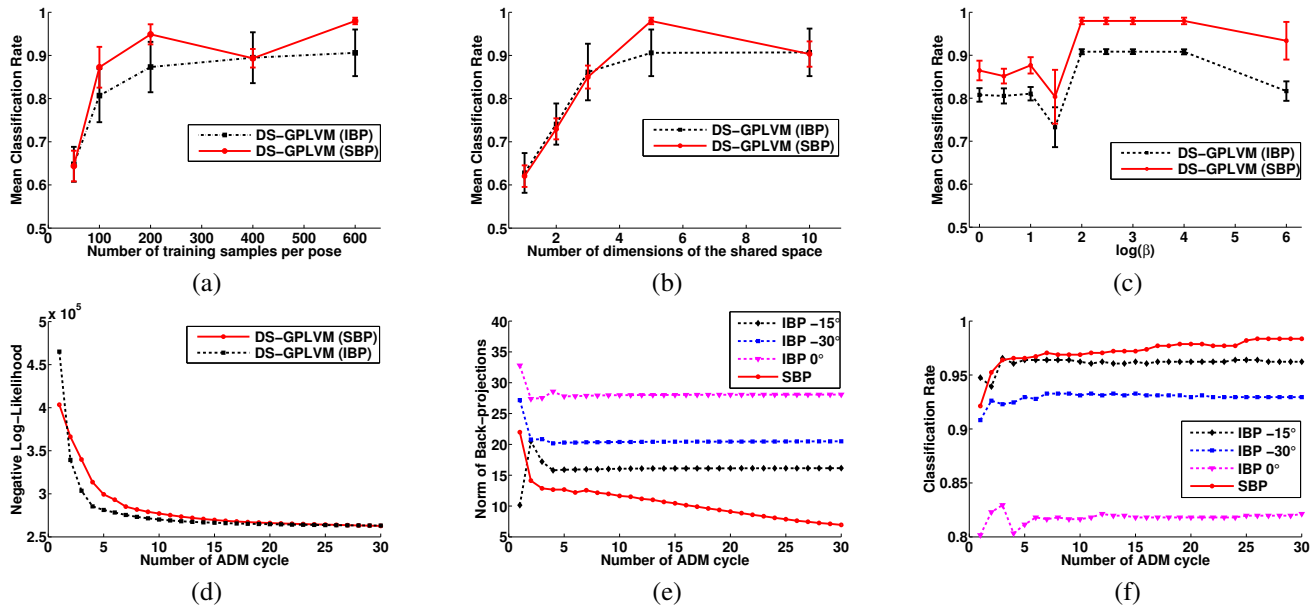


Fig. 3. DS-GPLVM. Upper row shows mean classification rate across all 5 poses from the MultiPIE dataset using feature-set (I) as a function of: (a) the number of training data per pose, (b) the dimensionality of the latent space, and (c) the prior scale parameter  $\beta$ . Lower row depicts: (d) the negative Log-Likelihood, (e) the norms of the constraints in the DS-GPLVM, and (f) the mean classification rate, as a function of the number of the ADM cycles.

more cycles of the ADM, the DS-GPLVM learns the shared manifold by taking into account all views, and thus, the error of back projections from the remaining views to the shared subspace decreases, while the one from the initialized view, *i.e.*, the  $-15^\circ$ , increases slightly – the consequence of the model trying to align the manifolds of different views. The red curve represents the error between the learned subspace and the back projections in the case of SBP. It is clear that the SBP variant outperforms the IBP variant of the model, since the former achieves a closer back-projection to the shared discriminative manifold, resulting in a better classification performance. This comes with a larger number of the ADM cycles during learning of the DS-GPLVM with SBP since it uses all views simultaneously to learn the back-mapping. Finally, from Figs. 3(e)-(f), we observe strong correlation between the norms of the model variants and the classification rate. In all cases, the increased classification performance is achieved by decreasing the gap between the shared-space and back-mappings, with both measures converging synchronously.

### C. Comparisons with other Multi-view Learning Methods

1) *Same Facial Features in Multiple Views*: We evaluate the proposed DS-GPLVM model across views in both view-invariant and multi-view setting. The former refers to the scenario where data from all views are used for training, while testing is performed using data from each view separately, and the latent space is back-constrained using the IBP. The latter refers to the scenario where data from all views are used during training and testing, and the latent space back-constrained using the SBP. The same strategy was used for evaluation of other multi-view techniques *i.e.*, GMLDA and GMLPP. Table I summarizes the results for the three sets of features, averaged across the five views from MultiPIE. We see that the facial

points (feature set (I)) result in a more discriminative descriptor for all methods, although we end up with higher standard deviation compared to the appearance features (feature sets (II) and (III)). Evidently, DS-GPLVM outperforms the other view-invariant and multi-view models on all three feature sets, showing that it can successfully unravel the discriminative shared-space that is better suited for FER. Interestingly, in this experiment LDA- and LPP-based linear methods achieve high accuracy, which is comparable to that of D-GPLVM and GPLRF. Moreover, GMLDA and GMLPP perform similarly to their single view trained counterparts, indicating that they were not able to fully benefit from the presence of additional views. We also observe a similar performance of the MvDA and the standard LDA. Note that, the accuracy of DS-GPLVM is higher by 3% than that of GPLRF, which is a special case of DS-GPLVM. We attribute this to the ability of the DS-GPLVM to integrate the discriminative information from multiple views into the shared space. We draw similar conclusions from the comparison between DS-GPLVM and DS-GPLVM (ind.), where the latter fails to impose the view constraints on the shared manifold.

Table II shows the performance of the models tested across all views, when feature set (I) (the best for all the models from Table I) is used. It is evident that the proposed DS-GPLVM performs consistently better than the compared models across all views. Note that all models achieve the lowest classification rate in the frontal view. However, the DS-GPLVM significantly improves the performance attained by the other models in this view. We attribute this to the fact that DS-GPLVM performs the classification in the shared space, where the classification of the expressions from the frontal view is facilitated due to the discriminative information learned from the other views. Furthermore, it is worth noting that the models' accuracy on the negative pan angles (the left side of the face) is

TABLE I

AVERAGE CLASSIFICATION RATE ACROSS FIVE VIEWS FROM THE MULTIPiE DATASET FOR THREE FEATURE SETS. IBP VERSION OF DS-GPLVM WAS TRAINED USING ALL AVAILABLE VIEWS, AND TESTED PER VIEW. THE REPORTED STANDARD DEVIATION IS ACROSS FIVE VIEWS.

Methods	Features		
	I	II	III
kNN	76.15 ± 5.42	81.71 ± 2.86	71.80 ± 2.23
LDA	87.72 ± 6.67	86.24 ± 2.31	87.02 ± 2.59
LPP	87.81 ± 6.65	86.16 ± 2.16	86.82 ± 2.60
D-GPLVM	87.17 ± 5.80	85.92 ± 2.95	86.87 ± 3.15
GPLRF	86.93 ± 6.30	85.58 ± 2.66	86.88 ± 2.91
GMLDA	86.72 ± 6.57	85.18 ± 2.94	86.40 ± 3.40
GMLPP	87.74 ± 6.12	86.10 ± 2.13	86.21 ± 2.06
MvDA	87.84 ± 6.51	86.66 ± 2.84	86.79 ± 2.86
DS-GPLVM (ind.)	88.64 ± 5.60	87.13 ± 2.73	87.34 ± 2.91
DS-GPLVM	<b>90.60 ± 5.40</b>	<b>88.44 ± 2.84</b>	<b>89.18 ± 2.83</b>

higher than on the corresponding positive pan angles (the right side of the face). Since MultiPIE contains more examples of negative emotion expressions, this confirms recent findings in [53] showing that the left hemisphere of the face is more informative when it comes to expressing negative emotions (*e.g.*, Disgust). The right hemisphere is more informative for positive emotions (*e.g.*, Happiness). In other words, due to the imbalance of the emotion categories in the used dataset, the learned classifiers were biased toward negative emotion expressions, and, hence, to the negative pan angles.

Table III compares the performance of the SBP variant of DS-GPLVM with other multi-view learning methods on three feature sets. The poor performance of KCCA can be attributed to its inherent propensity to overfitting training data, as also observed in, *e.g.*, [29]. In addition, both CCA and KCCA do not use any supervisory information during the subspace learning, which further explains their low performance. By comparing GPLRF (with concatenated features from different views) and DS-GPLVM, we see that the former, although not a multi-view method, performs comparably to our DS-GPLVM in the case of feature set (I). We attribute this to the fact that GPLRF can effectively explain variation in facial points from multiple views using a single GP. Yet, because of the large variation in the appearance of facial expressions from different views, the same is not the case when feature sets (II) and (III) are used. When compared to the state-of-the-art methods for multi-view learning (GMA and MvDA), DS-GPLVM performs similarly or better on all three feature sets. Furthermore, the SBP version of DS-GPLVM during inference succeeds to model complementary information from all available views, resulting in a higher accuracy compared to the best performing view, *i.e.*,  $-15^\circ$ , of the IBP variant of DS-GPLVM (see Table II).

2) *Feature Fusion*: We next evaluate DS-GPLVM in the feature fusion task, where the goal is to augment view-invariant facial expression classification by fusing different feature sets. Specifically, we trained the SBP version of DS-GPLVM using the three feature sets extracted from the frontal view only. This choice has been made because the frontal view is not the most informative one ( $-15^\circ$  is), and hence, there is a lot of space for improvement. From Table IV,

TABLE III

CLASSIFICATION RATE FOR THE MULTI-VIEW TESTING SCENARIO USING THE SBP VERSION OF DS-GPLVM. THE REPORTED STANDARD DEVIATION IS ACROSS THE 5 FOLDS.

Methods	Features		
	I	II	III
PW-CCA	72.42 ± 0.020	73.56 ± 0.025	56.07 ± 0.028
PW-KCCA	52.92 ± 0.039	69.15 ± 0.017	42.42 ± 0.026
GPLRF (conc.)	97.37 ± 0.014	89.42 ± 0.012	89.94 ± 0.012
GMLDA	96.33 ± 0.015	93.04 ± 0.011	92.15 ± 0.013
GMLPP	96.20 ± 0.014	91.37 ± 0.019	90.83 ± 0.017
MvDA	97.12 ± 0.017	93.56 ± 0.011	92.81 ± 0.015
DS-GPLVM	<b>97.98 ± 0.008</b>	<b>93.96 ± 0.015</b>	<b>93.29 ± 0.010</b>

TABLE IV

ACCURACY OF THE AUGMENTED CLASSIFICATION IN THE FRONTAL POSE. FEATURE FUSION IS ATTAINED WITH THE SBP VERSION OF DS-GPLVM.

Methods				
GPLRF (conc.)	GMLDA	GMLPP	MvDA	DS-GPLVM
83.16 ± 0.021	78.94 ± 0.018	85.95 ± 0.019	86.19 ± 0.014	<b>87.13 ± 0.019</b>



we see that the accuracy of DS-GPLVM in the frontal view outperforms that achieved by the GPLRF by more than 3%, where the features are simply concatenated and used as input. This is because GPLRF cannot fully account for variation in all three feature sets using a single GP. By contrast, DS-GPLVM learns separate GPs for each feature set, resulting in improved classification performance in the frontal view. It is also important to mention that by training GPLRF using each feature set separately, we obtained the following classification rates: 77.6%, 81.3% and 82.1%, for feature sets (I), (II), and (III), respectively. Compared to the accuracy of DS-GPLVM in Table IV (87.1%), the proposed feature fusion significantly outperforms each of the feature sets used independently. This is expected since the appearance features (LBPs and DCT), extracted from local patches, do not encode global information about face geometry, which is efficiently encoded by facial points. On the other hand, facial points are not informative about transient changes in facial appearance (*e.g.*, wrinkles and bulges) which are successfully captured by the appearance features. Thus, the combination of these features within the proposed framework turn out to be highly effective. The rest of multi-view methods also achieve significant increase in their performance (apart from GMLDA). However, DS-GPLVM outperforms (although marginally in some cases) all these state-of-the-art models.

3) *Same Facial Features in Different Illumination*: Herein, we evaluate the proposed DS-GPLVM under different illumination on MultiPIE, where the goal is to learn an illumination-free manifold for FER. For the purposes of this experiment, we used only images from the frontal view with two different lighting conditions: (i) no lighting source (dark view), and (ii) lighting from the flash of the corresponding camera (bright view). Each lighting condition has been considered as a separate view to train the IBP variant of DS-GPLVM with feature set III. DCT features were selected, since they are less robust to illumination variations than LBPs, and thus a difference in the performance between the two illumination conditions is

TABLE II  
VIEW-INVARIANT CLASSIFICATION RATE ON MULTIPIE DATASET FOR THE BEST FEATURE SET (*i.e.*, FACIAL POINTS (I)). IBP VERSION OF DS-GPLVM IS TRAINED USING ALL AVAILABLE VIEWS, AND TESTED PER VIEW. THE REPORTED STANDARD DEVIATION IS ACROSS 5 FOLDS.

Methods	Poses				
	$-30^\circ$	$-15^\circ$	$0^\circ$	$15^\circ$	$30^\circ$
kNN	80.88 $\pm$ 0.007	81.74 $\pm$ 0.014	68.36 $\pm$ 0.054	75.03 $\pm$ 0.024	74.78 $\pm$ 0.012
LDA	92.52 $\pm$ 0.015	94.37 $\pm$ 0.013	77.21 $\pm$ 0.014	87.07 $\pm$ 0.040	87.47 $\pm$ 0.007
LPP	92.42 $\pm$ 0.017	94.56 $\pm$ 0.011	77.33 $\pm$ 0.021	87.06 $\pm$ 0.045	87.68 $\pm$ 0.011
D-GPLVM	91.65 $\pm$ 0.017	93.51 $\pm$ 0.009	78.70 $\pm$ 0.021	85.96 $\pm$ 0.040	86.04 $\pm$ 0.010
GPLRF	91.65 $\pm$ 0.017	93.77 $\pm$ 0.007	77.59 $\pm$ 0.021	85.66 $\pm$ 0.026	86.01 $\pm$ 0.008
GMLDA	90.47 $\pm$ 0.012	94.18 $\pm$ 0.007	76.60 $\pm$ 0.029	86.64 $\pm$ 0.032	85.72 $\pm$ 0.015
GMLPP	91.86 $\pm$ 0.013	94.13 $\pm$ 0.002	78.16 $\pm$ 0.013	87.22 $\pm$ 0.023	87.36 $\pm$ 0.008
MvDA	92.49 $\pm$ 0.011	94.22 $\pm$ 0.014	77.51 $\pm$ 0.022	87.10 $\pm$ 0.031	87.89 $\pm$ 0.010
DS-GPLVM (ind.)	92.25 $\pm$ 0.013	94.83 $\pm$ 0.014	80.18 $\pm$ 0.025	87.63 $\pm$ 0.017	88.32 $\pm$ 0.023
DS-GPLVM	<b>93.55 <math>\pm</math> 0.019</b>	<b>96.96 <math>\pm</math> 0.012</b>	<b>82.42 <math>\pm</math> 0.018</b>	<b>89.97 <math>\pm</math> 0.023</b>	<b>90.11 <math>\pm</math> 0.028</b>

TABLE V  
CLASSIFICATION RATE ON THE FRONTAL VIEW UNDER DIFFERENT ILLUMINATION FOR FEATURE SET (III). THE IBP VARIANT OF DS-GPLVM WAS USED. THE REPORTED STANDARD DEVIATION IS ACROSS THE 5 FOLDS.

Methods	Illumination	
	Frontal flash 	No flash 
GPLRF	82.09 $\pm$ 0.015	77.00 $\pm$ 0.025
GMLDA	82.76 $\pm$ 0.017	84.01 $\pm$ 0.029
GMLPP	82.10 $\pm$ 0.029	84.75 $\pm$ 0.030
MvDA	83.80 $\pm$ 0.015	84.20 $\pm$ 0.019
DS-GPLVM	<b>85.51 <math>\pm</math> 0.032</b>	<b>85.68 <math>\pm</math> 0.021</b>

expected. From Table V we see that this difference is present in the results of the single-view method, *i.e.*, the GPLRF. The latter was trained separately for each lighting condition, and hence, the two learned manifolds falsely encoded the illumination as important information, resulting in a considerable gap between the performance of the bright and the dark view. Contrary to that, the compared multi-view methods, *i.e.*, GMLDA, GMLPP and MvDA, managed to remove, to some extent, the lighting condition of the views under the common space. This is evidenced by the improvement on the performance of the dark view, although a notable difference between the performance of the two views still exists. On the other hand, the proposed DS-GPLVM, not only achieved better results under both illumination conditions, but it also managed to align them by discarding the illumination under the shared space. Note that the DS-GPLVM reports similar classification rate, regardless the original lighting condition of the view.

#### D. Comparisons with other Multi-view Methods

We compare DS-GPLVM (with the IBP variant using feature set (III)) to the state-of-the-art methods for view-invariant FER. The results for the LGBP-based method, where the LBP features are extracted from Gabor images, are obtained from [6]. For the method in [12], we extracted the Sparse SIFT (SSIFT) features from the same images we used from MultiPIE. In both of the aforementioned methods, the target features (LGBP and SSIFT) are extracted per-view, and then fed into the view-specific SVM classifiers. We also compared

our model to the Coupled GP (CGP) model [9], where first view-normalization is performed by projecting a set of facial points (feature set (I)) from non-frontal views to the canonical view. In our experiments with CGP, we set the canonical view to the most discriminative view among the positive pan angles (*i.e.*,  $15^\circ$ ). This was followed by classification using the SVM learned in this view. Table VI shows comparative results. We observe first that all methods (except [12]) achieve the best results for the  $15^\circ$  view, indicating that regardless of the method/features employed, this view is more discriminative (among the positive pan angles) for the target task. We also note that DS-GPLVM outperforms on average the other two methods, which are based on the appearance features. This difference is in part due to the features used and in part due to the fact that the methods in [6] and [12] both fail to model correlations between different views. By contrast, the CGP method accounts for the relations between the views in a pairwise manner, while DS-GPLVM and DS-GPLVM (ind.) do so for all the views simultaneously. However, the proposed DS-GPLVM shows superior performance to that of DS-GPLVM (ind.), which in turn, outperforms CGP. This is because CGP performs view alignment (i) directly in the observation space, and (ii) without using any discriminative criterion during this process. Thus, the effects of high-dimensional noise and the errors of view-normalization adversely affect its performance in the classification task. On the other hand, DS-GPLVM (ind.) aligns the views directly in the shared space optimized for expression classification, while the proposed DS-GPLVM imposes further constraints on the shared manifold, resulting in a better performance on the target task. This is also reflected in the confusion matrices in Fig. 4. Note that the main source of confusion are the facial expressions of *Disgust* and *Squint*. This is because they are characterized by similar facial changes in the region of the eyes. However, the proposed DS-GPLVM improves significantly the accuracy on *Squint*, compared to the other models.

#### E. Cross Dataset Experiments on MultiPIE and LFPW

In this section, we test the ability of DS-GPLVM (the IBP variant) to generalize to unseen real-world spontaneous data. To this end, we evaluate different models on the smile detection task, where the feature set (I) extracted from images from MultiPIE is used for training. Images from LFPW are used

DI	77.3	4.9	0.0	3.4	14.2	0.0	DI	67.3	12.4	0.4	3.0	16.5	0.1	DI	73.4	6.8	1.4	4.3	13.6	0.1	DI	73.7	4.0	3.2	3.4	14.4	1.0
NE	0.7	95.0	0.0	2.9	1.0	0.0	NE	4.0	84.9	0.0	4.7	5.8	0.2	NE	2.5	89.2	0.1	5.4	1.3	1.0	NE	2.6	79.4	0.5	7.9	6.7	2.7
SC	1.1	0.0	96.3	0.9	0.6	0.8	SC	0.1	0.0	96.9	1.1	1.1	0.6	SC	1.1	0.2	94.5	1.0	0.5	2.4	SC	2.1	0.1	87.8	0.9	0.3	8.5
SM	1.6	10.1	0.0	86.7	1.4	0.0	SM	1.1	7.0	0.4	89.3	1.2	0.8	SM	3.5	8.1	0.1	85.0	1.7	1.4	SM	2.5	7.4	0.8	81.5	4.9	2.5
SQ	17.8	5.0	0.1	3.4	73.3	0.0	SQ	18.0	7.2	0.0	4.9	69.7	0.0	SQ	27.4	13.7	1.0	12.2	45.1	0.5	SQ	16.6	8.4	0.8	7.3	65.3	1.3
SU	0.6	0.9	0.8	3.4	0.0	94.0	SU	0.3	1.8	0.8	8.2	0.8	88.0	SU	2.3	6.5	6.1	6.6	0.0	78.2	SU	1.7	1.0	5.4	3.4	0.8	87.4
	DI	NE	SC	SM	SQ	SU		DI	NE	SC	SM	SQ	SU		DI	NE	SC	SM	SQ	SU		DI	NE	SC	SM	SQ	SU

(a) DS-GPLVM

(b) CGP

(c) SSIFT

(d) LGBP

Fig. 4. Comparative confusion matrices for FER over all angles of view for the (a) DS-GPLVM, (b) CGP, (c) SSIFT and (d) LGBP.

TABLE VI

COMPARISON OF TESTED METHODS ON THE MULTIPIE DATABASE. THE IBP VERSION OF DS-GPLVM WITH FEATURE SET (III), OUTPERFORMS THE STATE-OF-THE-ART METHODS FOR VIEW-INVARIANT FER. THE REPORTED STANDARD DEVIATION IS ACROSS 5 FOLDS.

Methods	Poses		
	0°	15°	30°
LGBP [6]	82.1	87.3	75.6
SSIFT [12]	81.14 ± 0.009	79.25 ± 0.016	77.14 ± 0.019
CGP [9]	80.44 ± 0.017	86.41 ± 0.013	83.73 ± 0.019
DS-GPLVM (ind.)	83.73 ± 0.029	88.41 ± 0.014	87.69 ± 0.022
DS-GPLVM	<b>84.31 ± 0.025</b>	<b>89.21 ± 0.015</b>	<b>90.26 ± 0.025</b>

for testing. This is a rather challenging task mainly because the test images are captured in an uncontrolled environment, which is characterized by large variation in head-poses and illumination, and occlusions of parts of the face. Also, the models are trained using data of *posed* (deliberately displayed as opposed to spontaneous and “in the wild”) expressions, which can differ considerably in subtlety compared to the *spontaneous* expressions used for testing. The difficulty of the task is evidenced by the results in Table VII, where we observe a significant drop in accuracy of all methods. Furthermore, we observe that the most informative views for smile detection are the ones with positive degrees (the right side of the face). This, again, is for the reasons explained in Sec. V-C1. However, all methods attain the higher accuracy in the frontal pose. We attribute this to the fact that the faces with non-frontal poses do not exactly belong to the discrete set of poses, but rather a continuous range from 0° to ±30°. Thus, the accuracy of the pose registration significantly affects the performance of the models. Nevertheless, the proposed DS-GPLVM outperforms the other models by a large margin in all poses except −30°. To explain this, we checked the number of test examples of smiles in this pose, and found that only few were available (contrary to other poses, which contained far more examples). Therefore, the misclassification of some resulted in a significant drop in the performance of both DS-GPLVM and DS-GPLVM (ind.).

#### F. Expression Recognition on Real World Images from SFEW

Finally, we evaluate the models on the feature fusion task, where the features are extracted from images of spontaneously displayed facial expressions in real-world environment. Specifically, we used LPQ [50] and PHOG [51] features from expressive images from the SFEW dataset. Contrary to the

TABLE VII

SMILE DETECTION IN IMAGES FROM LFPW DATASET. THE METHODS WERE TRAINED ON MULTIPIE DATASET USING FEATURE SET (I). WE USED THE IBP VERSION OF DS-GPLVM FOR THE VIEW-INVARIANT FER.

Method	Poses				
	−30°	−15°	0°	15°	30°
GMLDA	69.00	43.00	80.94	55.76	76.00
GMLPP	<b>70.00</b>	47.50	81.25	57.58	79.66
MvDA	<b>70.00</b>	50.00	81.25	51.52	<b>80.00</b>
DS-GPLVM (ind.)	57.20	52.50	84.00	69.38	<b>80.00</b>
DS-GPLVM	55.33	<b>58.00</b>	<b>90.00</b>	<b>74.55</b>	<b>80.00</b>

cross-dataset evaluation from the previous section, here both training and testing are performed using real-world spontaneous expression data. Note that LPQ is a texture descriptor that captures local information over a neighborhood of pixels, resulting in its being robust to illumination changes. On the other hand, PHOG is a local descriptor which is capable of preserving the spatial layout of the local shapes in an image. Thus, we expect the fusion of these two to achieve improved performance on the target task. The provided images of SFEW were originally divided into two subject independent folds, and we report the average results over the folds.

Table VIII shows the results obtained for different methods. We used the SBP variant of the DS-GPLVM. As the baseline we use the results obtained by the database creators [48]. The authors used non-linear SVM classifier on the concatenation of the features to report the classification rate on the feature fusion task. We see that all employed multi-view learning methods outperform the baseline on average. This is due to their ability to effectively exploit the discriminative information embedded in both feature spaces. However, in most cases, the linear multi-view learning methods are outperformed by the proposed DS-GPLVM. We attribute this to the fact that the linear models are unable to fully unravel the non-linear discriminative manifold of the used feature spaces. By contrast, this is handled better by the non-linear mappings in the DS-GPLVM, resulting in its average performance being the best among the tested models. Note, however, that in the case of Surprise, Fear and Neutral, its performance is lower than that of the linear models. By inspecting the back-projected test examples of these two expressions on the shared manifold, we observed that Neutral was spread around other emotion categories. This is because the varying level of expressiveness of different subjects, resulting in examples of Neutral being

TABLE VIII  
CLASSIFICATION RATES PER EXPRESSION CATEGORY OBTAINED BY DIFFERENT MODELS TRAINED/TESTED USING THE SFEW DATASET.

	Anger	Disgust	Fear	Happiness	Neutral	Sadness	Surprise	Average
Baseline	23.00	13.00	13.90	29.00	23.00	17.00	13.50	18.90
GMLDA	23.21	17.65	<b>29.29</b>	21.93	25.00	11.11	10.99	19.90
GMLPP	16.07	21.18	27.27	39.47	20.00	19.19	<b>16.48</b>	22.80
MvDA	23.21	17.65	27.27	40.35	<b>27.00</b>	10.10	13.19	22.70
DS-GPLVM	<b>25.89</b>	<b>28.24</b>	17.17	<b>42.98</b>	14.00	<b>33.33</b>	10.99	<b>24.70</b>

categorized as other expressions with low-intensity levels. As for the Surprise and Fear, the learned shared manifold indicated overfitting of these expressions. This is mainly due to subject differences, which adversely affected the ability of the back-mappings to correctly map these expressions onto the shared manifold. Nevertheless, DS-GPLVM outperformed the rest of the models on the remaining expressions, with a considerable improvement on Disgust, Happiness and Sadness.

## VI. CONCLUSION

In this paper, we proposed the DS-GPLVM model for learning a discriminative shared manifold of facial expressions from multiple views, that is optimized for the expression classification. This model is a generalization of latent variable models for learning a discriminative subspace of a single observation space. As such, it presents a complete non-parametric multi-view learning framework that can instantiate the rest of the compared non-linear single-view methods (*i.e.* D-GPLVM[17] and GPLRF [36]). As evidenced by our results on posed and spontaneously displayed facial expressions, when compared to the state-of-the-art methods for supervised multi-view learning and facial expression recognition, modeling of the manifold shared across different views and/or features using the proposed framework considerably improves both multi- and per- view/feature classification of facial expressions.

## APPENDIX A DERIVATIVES

During the optimization, we need to update  $\mathbf{X}$  and  $\theta_s$  by solving the problem in Eq. (26). The latter is a sum of two terms, the negative log-likelihood given by Eq. (20), and the norm term which, for convenience, we denote as

$$\mathbf{C} = \frac{\mu_t}{2} \sum_{v=1}^V \|\text{IBP}(\mathbf{X}, \mathbf{A}_t^{(v)}) + \frac{\Lambda_t^{(v)}}{\mu_t}\|_F^2 \quad (35)$$

Because of the likelihood term, the defined problem does not have an exact solution, and thus, we need to apply the CG algorithm. Hence, we have to compute the gradients of Eq. (20),(35) w.r.t. the latent positions  $\mathbf{X}$  and the kernel parameters  $\theta_s$

- $\frac{\partial L_s}{\partial \mathbf{X}} = \sum_v \frac{\partial L^{(v)}}{\partial \mathbf{X}} + \beta \tilde{\mathbf{L}} \mathbf{X}$
- $\frac{\partial L_s}{\partial \theta_s} = \left[ \frac{\partial L^{(1)}}{\partial \theta^{(1)}} \quad \dots \quad \frac{\partial L^{(V)}}{\partial \theta^{(V)}} \right]^T$
- $\frac{\partial \mathbf{C}}{\partial \mathbf{X}} = \sum_v \mu_t (\mathbf{X} - \mathbf{A}_t^{(v)}) + \Lambda_t^{(v)}$
- $\frac{\partial \mathbf{C}}{\partial \theta_s} = \mathbf{0}$ .

The likelihood term  $L^{(v)}$  is a function of the kernel  $\mathbf{K}^{(v)}$ , thus, we need to apply the chain rule in order to find the derivatives w.r.t  $\mathbf{X}$  and  $\theta^{(v)}$

- $\frac{\partial L^{(v)}}{\partial x_{ij}} = \text{tr} \left[ \left( \frac{\partial L^{(v)}}{\partial \mathbf{K}^{(v)}} \right)^T \frac{\partial \mathbf{K}^{(v)}}{\partial x_{ij}} \right]$
- $\frac{\partial L^{(v)}}{\partial \theta_i^{(v)}} = \text{tr} \left[ \left( \frac{\partial L^{(v)}}{\partial \mathbf{K}^{(v)}} \right)^T \frac{\partial \mathbf{K}^{(v)}}{\partial \theta_i^{(v)}} \right]$
- $\frac{\partial L^{(v)}}{\partial \mathbf{K}_v} = \frac{D}{2} (\mathbf{K}^{(v)})^{-1} - \frac{1}{2} (\mathbf{K}^{(v)})^{-1} \mathbf{Y}_v \mathbf{Y}_v^T (\mathbf{K}^{(v)})^{-1}$ .

Finally, the derivatives of the selected kernel are

- $\frac{\partial k^{(v)}(\mathbf{x}_i, \mathbf{x}_j)}{\partial \theta_1^{(v)}} = \exp(-\frac{\theta_2}{2} \|\mathbf{x}_i - \mathbf{x}_j\|^2)$
- $\frac{\partial k^{(v)}(\mathbf{x}_i, \mathbf{x}_j)}{\partial \theta_2^{(v)}} = -\frac{\theta_1^{(v)}}{2} \|\mathbf{x}_i - \mathbf{x}_j\|^2 \exp(-\frac{\theta_2}{2} \|\mathbf{x}_i - \mathbf{x}_j\|^2)$
- $\frac{\partial k^{(v)}(\mathbf{x}_i, \mathbf{x}_j)}{\partial \theta_3^{(v)}} = 1$
- $\frac{\partial k^{(v)}(\mathbf{x}_i, \mathbf{x}_j)}{\partial \theta_4^{(v)}} = -\frac{1}{(\theta_4^{(v)})^2} \delta_{i,j}$

and

$$\frac{\partial \mathbf{k}^{(v)}(\mathbf{x}_i)}{\partial x_{ij}} = \begin{bmatrix} -\theta_2(x_{ij} - x_{1j}) k^{(v)}(\mathbf{x}_i, \mathbf{x}_1) \\ \vdots \\ -\theta_2(x_{ij} - x_{Nj}) k^{(v)}(\mathbf{x}_i, \mathbf{x}_N) \end{bmatrix}$$

## APPENDIX B

### LOO SOLUTION OF THE REGRESSION STEP IN ADM

Herein, we derive the solution for the more general form of the IBP case. The same steps can be followed to arrive at the solution of the SBP case. The optimal values of parameters  $\mathbf{A}^{(v)}$  are given by the solution of the linear equation:

$$(\mathbf{K}_{bc}^{(v)} + \frac{\lambda^{(v)}}{\mu_t} \mathbf{I}) \mathbf{A}^{(v)} = (\mathbf{X} + \frac{\Lambda_t^{(v)}}{\mu_t}). \quad (36)$$

The system of linear equations defined by Eq. (36) is insensitive to permutations of the ordering of the equations and the variables. Thus, at each iteration of the LOO, the  $i$ -th left out sample and the corresponding equation can be placed on top, without affecting the result. This enables us to define the matrix  $\mathbf{M}$  as in Eq. (31). By placing  $\mathbf{M}$  back in Eq. (36), we end up with the following linear system of equations:

$$\begin{bmatrix} m_{ii} & \mathbf{m}_i^T \\ \mathbf{m}_i & \mathbf{M}_i \end{bmatrix} \mathbf{A}^{(v)} = \begin{bmatrix} \mathbf{x}_i + \Lambda_i^{(v)} / \mu_t \\ \mathbf{X}^{(-i)} + \Lambda_{-i}^{(v)} / \mu_t \end{bmatrix} \quad (37)$$

Now, the solution of the parameters of the regression with the  $i$ -th sample excluded is

$$\mathbf{A}_{-i}^{(v)} = \mathbf{M}_i^{-1} (\mathbf{X}^{(-i)} + \frac{\Lambda_{-i}^{(v)}}{\mu_t}),$$

and the LOO prediction of the  $i$ -th sample is given by

$$\begin{aligned}\hat{\mathbf{x}}_i^{(-i)} &= \mathbf{m}_i^T \mathbf{A}_{-i}^{(v)} = \mathbf{m}_i^T \mathbf{M}_i^{-1} (\mathbf{X}^{(-i)} + \frac{\boldsymbol{\Lambda}_{-i}^{(v)}}{\mu_t}) \\ &= \mathbf{m}_i^T \mathbf{M}_i^{-1} [\mathbf{m}_i \quad \mathbf{M}_i] \mathbf{A}^{(v)} \\ &= \mathbf{m}_i^T \mathbf{M}_i^{-1} [\mathbf{m}_i \quad \mathbf{M}_i] \begin{bmatrix} \mathbf{A}_i^{(v)} \\ \mathbf{A}_{-i}^{(v)} \end{bmatrix} \\ &= \mathbf{m}_i^T \mathbf{M}_i^{-1} \mathbf{m}_i \mathbf{A}_i^{(v)} + \mathbf{m}_i^T \mathbf{A}_{-i}^{(v)}.\end{aligned}$$

From Eq. (37) we have

$$\mathbf{x}_i + \frac{\boldsymbol{\Lambda}_i^{(v)}}{\mu_t} = [m_{ii} \quad \mathbf{m}_i^T] \begin{bmatrix} \mathbf{A}_i^{(v)} \\ \mathbf{A}_{-i}^{(v)} \end{bmatrix} = m_{ii} \mathbf{A}_i^{(v)} + \mathbf{m}_i^T \mathbf{A}_{-i}^{(v)} \quad (38)$$

and thus, the error between the prediction  $\hat{\mathbf{x}}_i^{(-i)}$  and the actual output  $\mathbf{x}_i$  is

$$\begin{aligned}\mathbf{x}_i - \hat{\mathbf{x}}_i^{(-i)} &= (m_{ii} - \mathbf{m}_i^T \mathbf{M}_i^{-1} \mathbf{m}_i) \mathbf{A}_i^{(v)} - \boldsymbol{\Lambda}_i^{(v)} / \mu_t \\ &= \frac{\mathbf{A}_i^{(v)}}{[\mathbf{M}^{-1}]_{ii}} - \frac{\boldsymbol{\Lambda}_i^{(v)}}{\mu_t},\end{aligned}$$

where on the last equation we used the Shur complement from the block matrix inversion lemma, and  $\mathbf{M}_{ii}$  denotes the  $i$ -th diagonal element of the matrix  $\mathbf{M}$ . Finally, we end up with the cost of the LOO for all samples,  $E_{LOO}$ , as defined in Eq. (34). For the SBP case we follow exact the same steps, with the difference that we drop from all the equations the dependencies on the view  $v$  and we replace the  $\mathbf{K}_{bc}^{(v)}$  with

$$\tilde{\mathbf{K}} = \sum_{v=1}^V w_v \mathbf{K}_{bc}^{(v)}.$$

Our final goal is to find the optimal parameters  $\gamma^{(v)}$  and  $\lambda^{(v)}$  that minimize the error of the LOO cross validation, defined by Eq. (34). For this, we need to calculate the derivatives of  $E_{LOO}$  w.r.t.  $\gamma^{(v)}$  and  $\lambda^{(v)}$ . We first define the diagonal matrix

$$\mathbf{D} = \begin{bmatrix} \frac{1}{[\mathbf{M}^{-1}]_{11}} & & \\ & \ddots & \\ & & \frac{1}{[\mathbf{M}^{-1}]_{NN}} \end{bmatrix}$$

that allows us to reformulate Eq. (34) into

$$E_{LOO} = \frac{1}{2} \left\| \mathbf{D} \mathbf{A}^{(v)} - \frac{\boldsymbol{\Lambda}^{(v)}}{\mu_t} \right\|^2. \quad (39)$$

Using the chain rule, the derivatives of Eq. (39) are given by

$$\frac{\partial E_{LOO}}{\partial \lambda^{(v)}} = \text{tr} \left[ \left( \frac{\partial E_{LOO}}{\partial \mathbf{A}^{(v)}} \right)^T \frac{\partial \mathbf{A}^{(v)}}{\partial \lambda^{(v)}} + \left( \frac{\partial E_{LOO}}{\partial \mathbf{D}} \right)^T \frac{\partial \mathbf{D}}{\partial \lambda^{(v)}} \right]$$

and

$$\frac{\partial E_{LOO}}{\partial \gamma^{(v)}} = \text{tr} \left[ \left( \frac{\partial E_{LOO}}{\partial \mathbf{A}^{(v)}} \right)^T \frac{\partial \mathbf{A}^{(v)}}{\partial \gamma^{(v)}} + \left( \frac{\partial E_{LOO}}{\partial \mathbf{D}} \right)^T \frac{\partial \mathbf{D}}{\partial \gamma^{(v)}} \right],$$

while the detailed derivatives inside the trace terms are

- $\frac{\partial E_{LOO}}{\partial \mathbf{A}^{(v)}} = \mathbf{D}^T (\mathbf{D} \mathbf{A}^{(v)} - \frac{\boldsymbol{\Lambda}^{(v)}}{\mu_t})$
- $\frac{\partial E_{LOO}}{\partial \mathbf{D}} = \left[ \mathbf{D} \mathbf{A}^{(v)} (\mathbf{A}^{(v)})^T - \frac{1}{\mu_t} \boldsymbol{\Lambda}^{(v)} (\mathbf{A}^{(v)})^T \right] \odot \mathbf{I}$
- $\frac{\partial \mathbf{A}^{(v)}}{\partial \lambda^{(v)}} = -\mathbf{M}^{-1} \frac{\partial \mathbf{M}}{\partial \lambda^{(v)}} \mathbf{M}^{-1} (\mathbf{X} + \frac{\boldsymbol{\Lambda}_t^{(v)}}{\mu_t}) = -\frac{1}{\mu_t} \mathbf{M}^{-1} \mathbf{A}^{(v)}$

- $\frac{\partial \mathbf{A}^{(v)}}{\partial \gamma^{(v)}} = -\mathbf{M}^{-1} \frac{\partial \mathbf{M}}{\partial \gamma^{(v)}} \mathbf{M}^{-1} (\mathbf{X} + \frac{\boldsymbol{\Lambda}_t^{(v)}}{\mu_t}) = -\mathbf{M}^{-1} \frac{\partial \mathbf{K}_{bc}^{(v)}}{\partial \gamma^{(v)}} \mathbf{A}^{(v)}$
- $\frac{\partial \mathbf{D}}{\partial \lambda^{(v)}} = -(\mathbf{D} \odot \mathbf{D}) \odot \frac{\partial \mathbf{M}^{-1}}{\partial \lambda^{(v)}} = (\mathbf{D} \odot \mathbf{D}) \odot (\mathbf{M}^{-1} \mathbf{M}^{-1})$
- $\frac{\partial \mathbf{D}}{\partial \gamma^{(v)}} = -(\mathbf{D} \odot \mathbf{D}) \odot \frac{\partial \mathbf{M}^{-1}}{\partial \gamma^{(v)}} = (\mathbf{D} \odot \mathbf{D}) \odot (\mathbf{M}^{-1} \frac{\partial \mathbf{K}_{bc}^{(v)}}{\partial \gamma^{(v)}} \mathbf{M}^{-1})$

where the value of  $\frac{\partial \mathbf{K}_{bc}^{(v)}}{\partial \gamma^{(v)}}$  for each element of the kernel is given in Appendix A and  $\odot$  denotes the Hadamard product of two matrices. Once we have obtained the optimal parameters  $\gamma^{(v)}$  and  $\lambda^{(v)}$ , we can compute  $\mathbf{A}^{(v)}$  from Eq. (36).

#### ACKNOWLEDGMENT

This work has been funded by the European Community 7th Framework Programme [FP7/2007-2013] under grant agreement no. 611153 (TERESA).

#### REFERENCES

- [1] M. Pantic, A. Nijholt, A. Pentland, and T. Huanag, "Human-Centred Intelligent Human-Computer Interaction (HCI<sup>2</sup>): how far are we from attaining it?" *International Journal of Autonomous and Adaptive Communications Systems*, vol. 1, no. 2, pp. 168–187, 2008.
- [2] A. Vinciarelli, M. Pantic, and H. Bourlard, "Social signal processing: Survey of an emerging domain." *Image and Vision Computing*, vol. 27, no. 12, pp. 1743–1759, 2009.
- [3] Z. Zeng, M. Pantic, G. Roisman, and T. Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 39–58, 2009.
- [4] Z. Zhu and Q. Ji, "Robust real-time face pose and facial expression recovery," in *IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 681–688.
- [5] P. Ekman and W. V. Friesen, *Unmasking the face: A guide to recognizing emotions from facial clues*. Los Altos, CA, USA: Ishk, 2003.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] O. Rudovic, M. Pantic, and I. Patras, "Coupled Gaussian processes for pose-invariant facial expression recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1357–1369, 2013.
- [10] O. Rudovic, I. Patras, and M. Pantic, "Regression-based multi-view facial expression recognition," in *Proceedings of Int'l Conf. Pattern Recognition (ICPR'10)*, Istanbul, Turkey, August 2010, pp. 4121–4124.
- [11] W. Zheng, H. Tang, Z. Lin, and T. Huang, "Emotion recognition from arbitrary view facial images," *European Conf. on Computer Vision*, pp. 490–503, 2010.
- [12] U. Tariq, J. Yang, and T. Huang, "Multi-view facial expression recognition analysis with generic sparse coding feature," in *European Conf. on Computer Vision (ECCV-W'12)*, 2012, pp. 578–588.
- [13] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs, "Generalized multiview analysis: A discriminative latent space," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2012, pp. 2160–2167.
- [14] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," in *European Conf. on Computer Vision*, 2012, pp. 808–821.
- [15] A. Shon, K. Grochow, A. Hertzmann, and R. Rao, "Learning shared latent structure for image synthesis and robotic imitation," *Advances in Neural Information Processing Systems*, vol. 18, p. 1233, 2006.
- [16] C. Ek and P. Lawrence, "Shared Gaussian process latent variable models," Ph.D. dissertation, Oxford Brookes University, 2009.
- [17] R. Urtasun and T. Darrell, "Discriminative Gaussian process latent variable model for classification," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 927–934.

- [18] D. P. Bertsekas, "Constrained optimization and Lagrange multiplier methods," *Computer Science and Applied Mathematics, Boston: Academic Press, 1982*, vol. 1, 1982.
- [19] S. Eleftheriadis, O. Rudovic, and M. Pantic, "Shared Gaussian Process Latent Variable Model for Multi-view Facial Expression Recognition," in *ISVC, 2013*, pp. 527–538.
- [20] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
- [21] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.
- [23] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.
- [24] T. F. Cootes, G. J. Edwards, C. J. Taylor *et al.*, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001.
- [25] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Transactions on Computers*, vol. 100, no. 1, pp. 90–93, 1974.
- [26] D. Lowe, "Object recognition from local scale-invariant features," in *Int'l Conf. on Computer Vision*, vol. 2, 1999, pp. 1150–1157.
- [27] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2009, pp. 1794–1801.
- [28] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.
- [29] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, 2004.
- [30] J. Rupnik and J. Shawe-Taylor, "Multi-view canonical correlation analysis," in *Conf. on Data Mining and Data Warehouses*, 2010, pp. 1–4.
- [31] A. Kumar and H. D. Iii, "A co-training approach for multi-view spectral clustering," in *Proc. of the Int'l Conf. on Machine Learning*, 2011, pp. 393–400.
- [32] T. Diethe, D. R. Hardoon, and J. Shawe-Taylor, "Constructing nonlinear discriminants from multiple data views," in *Machine learning and knowledge discovery in databases*. Springer, 2010, pp. 328–343.
- [33] C. Bishop, *Pattern recognition and machine learning*. Springer New York, 2006, vol. 4.
- [34] X. Niyogi, "Locality preserving projections," in *Neural information processing systems*, vol. 16, 2004, p. 153.
- [35] N. Lawrence, "Probabilistic non-linear principal component analysis with Gaussian process latent variable models," *The Journal of Machine Learning Research*, vol. 6, pp. 1783–1816, 2005.
- [36] G. Zhong, W.-J. Li, D.-Y. Yeung, X. Hou, and C.-L. Liu, "Gaussian process latent random field," in *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010, pp. 679–684.
- [37] C. Rasmussen and C. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 1.
- [38] N. D. Lawrence and J. Q. Candela, "Local distance preservation in the GP-LVM through back constraints," in *Int'l Conf. in Machine Learning*, vol. 148. ACM, 2006, pp. 513–520.
- [39] F. R. Chung, "Spectral graph theory," *American Mathematical Society*, 1997.
- [40] H. Rue and L. Held, *Gaussian Markov random fields: theory and applications*. Chapman & Hall, 2005, vol. 104.
- [41] M. Salzmann and R. Urtasun, "Implicitly constrained Gaussian process regression for monocular non-rigid pose estimation," in *Advances in Neural Information Processing Systems*, 2010, pp. 2065–2073.
- [42] X. Zhu, J. Lafferty, and Z. Ghahramani, "Semi-supervised learning: from Gaussian fields to Gaussian processes," School of CS, CMU, Tech. Rep. CMU-CS-03-175, 2003.
- [43] J. Hainmueller and C. Hazlett, "Kernel regularized least squares: reducing misspecification bias with a flexible and interpretable machine learning approach," *Political Analysis*, vol. 22, no. 2, pp. 143–168, 2014.
- [44] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Computational learning theory*. Springer, 2001, pp. 416–426.
- [45] S. Sundararajan and S. S. Keerthi, "Predictive approaches for choosing hyperparameters in Gaussian processes," *Neural Computation*, vol. 13, no. 5, pp. 1103–1118, 2001.
- [46] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.
- [47] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2011, pp. 545–552.
- [48] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon, "Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark," in *Int'l Conf on Computer Vision Workshops*, 2011, pp. 2106–2112.
- [49] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "A semi-automatic methodology for facial landmark annotation," in *5th Workshop on AMFG, Proc. of the Int'l Conf. CVPR-W13*, 2013.
- [50] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *Image and Signal Processing*. Springer, 2008, pp. 236–243.
- [51] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *Proceedings of the Int'l Conf. on Image and Video Retrieval*, 2007, pp. 401–408.
- [52] Z. Zheng, F. Yang, W. Tan, J. Jia, and J. Yang, "Gabor feature-based face recognition using supervised locality preserving projection," *Signal Processing*, vol. 87, no. 10, pp. 2473–2483, 2007.
- [53] M. Pantic and I. Patras, "Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences," *IEEE Transactions on Systems, Man and Cybernetics - Part B*, vol. 36, no. 2, pp. 433–449, 2006.



**Stefanos Eleftheriadis** received a Diploma in Electrical 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. He currently, pursues a PhD degree in the Computing Department, Imperial College London, U.K. His research interests are in automatic human behavior analysis, machine learning and computer vision.



**Ognjen Rudovic** received his PhD from Imperial College London, Computing Dept., UK, in 2014, and a MSc degree in Computer Vision and Artificial Intelligence from Computer Vision Center (CVC), Spain, in 2008. He is currently working as a Research Associate at the Computing Department, Imperial College London, UK. His research interests are in automatic recognition of human affect, machine learning and computer vision.



**Maja Pantic** is Professor in Affective and Behavioural Computing at Imperial College London, Computing Dept., UK, and at the University of Twente, Dept. of Computer Science, Netherlands. She 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 currently serves as the Editor in Chief of Image and Vision Computing Journal, and as an Associate Editor of IEEE Trans. on Systems, Man, and Cybernetics Part B and IEEE TPAMI.