Leveraging the Deep Learning Paradigm for Continuous Affect Estimation from Facial Expressions

Meshia Cédric Oveneke, Yong Zhao, Ercheng Pei, Dongmei Jiang, and Hichem Sahli

Abstract—Continuous affect estimation from facial expressions has attracted increased attention in the affective computing research community. This paper presents a principled framework for estimating continuous affect from video sequences. Based on recent developments, we address the problem of continuous affect estimation by leveraging the Bayesian filtering paradigm, i.e. considering affect as a latent dynamical system corresponding to a general feeling of pleasure with a degree of arousal, and recursively estimating its state using a sequence of visual observations. To this end, we advance the state-of-the-art as follows: (i) Canonical face representation (CFR): a novel algorithm for two-dimensional face frontalization, (ii) Convex unsupervised representation learning (CURL): a novel frequency-domain convex optimization algorithm for unsupervised training of deep convolutional neural networks (CNN)s, and (iii) Deep extended Kalman filtering (DEKF): an extended Kalman filtering-based algorithm for affect estimation from a sequence of deep CNN observations. The performance of the resulting CFR-CURL-DEKF algorithmic framework is empirically evaluated on publicly available benchmark datasets for facial expression recognition (CK+) and continuous affect estimation (AVEC 2012 and AVEC 2014).

Index Terms—Affect Estimation, Facial Expressions, Face Frontalization, Partial Least-Squares Regression, Convolutional Auto-Encoders, Extended Kalman Filtering

1 Introduction

Automated analysis of human affective behavior has attracted increased attention from researchers in affective computing. Among the different types of affective behaviors, particular attention has been paid to facial expressions and its automated analysis from video [1]. At the heart of facial expression analysis are a set of complex information processing challenges due to the complex deformation of the face, the loss of 3D information during the image formation process [2] and the presence of nuisance factors such as person-specific morphology, view-point variations, unknown lighting conditions [3] and lack of reliable ground-truth data [4], [5]. In addition to these technical challenges, the lack of agreement on affect-related terminology has spilled over into more fundamental research areas such as computer vision and machine learning. We therefore argue that there is a growing need for a principled framework for designing affective computing systems capable of coping with the above-mentioned issues.

In this work, we propose a principled framework for continuous estimation of human affect given an incoming stream of image sequences containing facial displays (expressions) thereof. Our proposed framework is inspired by our recent work [6], where we proposed to leverage the Bayesian filtering paradigm [7] for continuous affect estimation, by considering affect as a continuous state corresponding to a general feeling of pleasure-displeasure with some degree of arousal [8]. We further assumed that the affective state and its temporal evolution is governed by "laws" and, consequently, posed the problem of affect estimation as a problem of estimating the latent state of a dynamical system based on a sequence of noisy measurements related to the state of the system. This way of posing the problem opened new horizons and has been successfully applied to the problem of speech-based emotion prediction [9]. Following the much celebrated three levels of understanding for complex information systems introduced by David Marr [10], we can consider that our previous work [6] addressed the first level: *computational theory*, i.e. what is the goal of the computation, why is it appropriate, and what is the logic of the strategy by which it can be carried out? In that sense, the present paper complements our previous work and addresses the second level of understanding: *representation and algorithm*, i.e. how can the computational theory be implemented? In particular, what is the representation for the input and output, and what is the algorithm for the transformation? As answer to these questions, we propose to leverage the *deep learning* (DL) paradigm, motivated by its capacity to represent information and implement various computational tasks [11].

In the context of continuous affect estimation from facial expressions, our proposed approach is three-fold: In a

- *M. C. Oveneke, Y. Zhao and E. Pei are with the Vrije Universiteit Brussel (VUB), Dept. of Electronics & Informatics (ETRO), VUB-NPU joint AVSP Research Lab. Pleinlaan 2, 1050 Brussels, Belgium. E-mail: {mcovenek,yzhao,epei}@etrovub.be*
- *D. Jiang is with the Northwestern Polytechnical University (NPU), Shaanxi Key Lab on Speech and Image Information Processing, VUB-NPU joint AVSP Research Lab. Youyo Xilu 127, Xi'an 710072, China. E-mail: jiangdm@nwpu.edu.cn*
- *H. Sahli is with the Vrije Universiteit Brussel (VUB), Dept. of Electronics & Informatics (ETRO), VUB-NPU joint AVSP Research Lab. Pleinlaan 2, 1050 Brussels, Belgium, and with the Interuniversity Microelectronics Centre (IMEC), Kapeldreef 75, 3001 Heverlee, Belgium. E-mail: hsahli@etrovub.be*

Manuscript received February 13, 2016; revised September 30, 2016.

first stage, we propose a simple method for canonical face representation for discarding nuisance factors such as head size, pose and, up to a certain extent, identity. The proposed method consists of a *partial least squares* (PLS) regression based face frontalization algorithm, where PLS is used for estimating the neutral (non-expressive) 2D face geometry, followed by piece-wise affine texture warping. In a second stage, we build upon our recent work on spectral convolutional auto-encoders [12] and propose an efficient algorithmic framework for greedy layer-wise unsupervised learning of deep *convolutional neural networks* (CNN)s on top of the PLS based canonical face representations. The choice of auto-encoder based greedy layer-wise training is motivated by the observation that certain types of auto-encoders such as *reconstruction contractive auto-encoders* (RCAE)s are able to capture the factors of variations of the data-generating distribution [13]. From an auto-encoder point of view, our technical contribution consists of randomly fixing the (non-linear) encoding parameters and only training the (linear) decoding parameters, yielding an easy-to-solve convex optimization problem, hence *convex unsupervised representation learning* (CURL). In a third and final stage, we use the representation outputted from the top-layer of deep CNNs as observation for *Kalman filtering* (KF) based continuous affect estimation. We augment the linear KF solution presented in [6] by modeling the state transition and observation models using *neural networks* (NN)s, hence *deep extended Kalman filter* (DEKF).

The outline of the paper is as follows. Section 2 gives a description of the background related to facial expressions and affect, and provides an overview of related work using deep learning techniques. In section 3, we give a formal definition of the continuous affect estimation problem and further give a detailed description of our deep learning based solution to the problem. Section 4 assesses the performance of our proposed model in terms of experimental results on benchmark datasets for facial expression recognition on the *extended Cohn-Kanade* (CK+) dataset [14] and continuous affect estimation on the *audio-visual emotion challenge* (AVEC) 2012 and 2014 datasets [15], [16]. Section 5 discusses our main findings and concludes our work.

2 Background and Related Work

2.1 Facial Expressions and Continuous Affect

The scientific study of facial expressions has been pursued now for about two centuries. Although it is widely accepted that facial expressions are vital for communicating social signals, it is still unclear what affective information they convey. In fact, some have been arguing there is no evidence to support a link between what appears on someone's face and how they feel inside [17], [18]. In the following, we briefly summarize the notions of facial expressions, continuous affect and their relation.

2.1.1 Facial Expressions

From an information-processing perspective, facial expressions consist of a physical component and an affective component [19]. On one hand, the physical component of facial expressions is well understood and is known to consist of observable morphological changes caused by facial muscle movements. On the other hand, the affective component is not well understood but has for a long time been assumed to reflect a person's internal feelings and basic emotions such as "happiness", "anger", "sadness", "fear", "disgust" or "surprise", which are assumed to be culturally universal and rooted in biological adaptive functions [20], [21]. It is now a consensus that facial expressions are not mapped to emotions in a one-to-one manner, but are (partially) caused by affect. The following paragraph clarifies this notion.

2.1.2 Continuous Affect

Common sense tells us that emotions are expressed on the face and easily decoded by a perceiver without the use of language. This would imply that certain combinations of facial muscle movements encode emotions in a predictable manner. Meanwhile, increasing research results have demonstrated that people do not consistently produce the specific configurations of facial muscle movements predicted by basic emotions [17]. In other words, basic emotions are not evidenced as consistent and specific patterns in facial muscle movements. Moreover, there is no convincing evidence of a unique pattern for each emotion in the autonomic nervous system [22]. In contrast, studies find evidence that facial muscle movements consistently correspond to a general feeling of pleasure–displeasure with some degree of arousal [23].

The general feeling of pleasure–displeasure with some degree of arousal is called *"core" affect* or simply *affect*, which has been defined as [8]: *"A neuro–physiological state that is consciously accessible as a simple, non-reflective feeling that is an integral blend of hedonic (pleasure-displeasure) and arousal (sleepy-activated) values."*. In that sense, we can loosely associate the neuro-physiological component to an unconscious experience and the psychological component to a conscious experience. Affect, when changing rapidly and directed at an object and accompanied by certain cognitions, physiological and behavioral changes, is what we know as "basic" emotions. In that sense, the core affect model suggests that "basic" emotions are not biologically hard-wired, but instead, are phenomena that emerge in consciousness "in the moment".

2.2 Leveraging the Deep Learning Paradigm

It is widely accepted that most of the quantitative improvements to computer vision tasks obtained in the past dozen years can be ascribed to the introduction of improved representations [24]. Among the most successful approaches is the much-celebrated deep learning. While substantial progress has been achieved with deep learning for tasks such as face recognition and verification, the adoption of this technique has been relatively slow in the field of facial affective computing. The main advantage of deep learning approaches is the general-purpose capability of modeling image representation (local-to-global) as well as the recognition or estimation process in a parametric and data-driven manner. This potential end-to-end and data-driven strategy is what makes deep learning appealing for tackling complex computer vision tasks, without the need for axiomatic knowledge. In the following, we give an overview of how deep learning approaches have been

applied to the problems of facial expression recognition and facial affect estimation.

2.2.1 Deep Learning for Facial Expression Recognition

One of the first exploitation of deep learning in the context of facial expression analysis was presented in [25]. In this work, the authors introduced a semi-supervised deep learning approach for facial expression recognition by separating emotion related features from the unrelated ones. As a deep learning approach, a *convolutional contractive auto-encoder* (CCAE) layer and max-pooling layer were used for the feature extraction stage. In [26], a convolutional layer and max-pooling layer were first used to extract over-complete features for the global appearance variation. Then the so-called AU-aware receptive field layer was connected as a supervised feature selection scheme to extract mid-level features. Finally, a multi-layer *restricted Boltzmann machine* (RBM) [27], [28], [29] was added to learn higher level hierarchical features. In [30], a multi-layer *boosted deep belief network* (BDBN) was proposed by integrating feature learning, feature selection and classification uniformly in one framework for the first time.

More recent work have exploited fully supervised deep learning approaches. In [31], a zero-bias deep CNN was trained from scratch with three convolutional layers, one fully connected layer and a softmax layer, achieving state-of-the-art results on the CK+ dataset. To alleviate overfitting issues, some recent works resort to transferring the prior face knowledge from a pre-trained face recognition deep network to the facial expression network. In [32], the *peak-piloted deep network* (PPDN) was proposed to perform intensity invariant facial expression recognition by transforming the non-peak expression features to peak expression features. They used the GoogLeNet [33] as pre-trained network architecture before fine-tuning. In [34], the authors introduced a novel learning algorithm that utilizes the pre-trained face recognition *vision geometry group* (VGG) network [33] to regularize the training of the expression recognition network. In summary, although these deep learning methods have improved the state-of-the-art performance, they often proposed a complicated or very deep model structure, thus heavily rely on well-annotated large datasets and high computational resources, which is a serious bottleneck for applying deep learning to facial expression analysis.

2.2.2 Deep Learning for Facial Affect Estimation

Although deep learning has been successfully applied to facial expression recognition, the task of continuous affect estimation is more challenging, mainly due to the inherently temporal nature of the problem and the often unreliable (i.e. uncertain and misaligned) continuous-valued annotations. Inspired by deep learning approaches used for facial expression recognition, the majority of deep learning techniques applied for affect estimation revolve around learning static discriminative templates via deep CNNs. The work in [35] introduced a technique for the discriminative training of deep *recurrent neural networks* (RNN)s [36] using the *concordance correlation coefficient* (CCC) as cost function, which unites both correlation and mean squared error in a single differentiable function. In [37] and [38], a CNN was trained frame-by-frame based on the classical *mean squared error* (MSE) objective. In order to incorporate temporal information, they used the extracted CNN features for each video frame and concatenated the resulting recent several frames feature representation from the last hidden layer as a frame input to a *long short-term memory* (LSTM)-RNN [39]. In other works such as [40] and [38], a *deep residual network* (ResNet) [41] was used as deep learning architecture. In [40], the last pooling layer was used as feature extractor and as input to a *support vector regressor* (SVR) for frame-based affect estimation. In [42], the authors compared the performance of two state-of-the-art machine learning techniques, namely the *bi-directional long short-term memory recurrent neural networks* (BLSTM-RNN) and SVR for the task of valence-arousal prediction. This work showed that, on average, BLSTM-RNNs outperform SVR due to their ability to model temporal information.

The above overview shows that most of the deep learning approaches adopted so far do not take the dynamics of affect into account. Most of them even rely on (pre)training on a facial expression recognition task before applying to affect estimation. We therefore argue that a more appropriate application of deep learning models/algorithms is needed for addressing the specific challenges of the affect estimation task. This motivates us to take a closer look and apply deep learning in a different and more appropriate way, i.e. through the Bayesian filtering paradigm. To the best of our knowledge, the present paper is one of the first that exploits the synergy between deep learning and Bayesian filtering for addressing the affect estimation problem. In the following, we formally pose our problem in terms of computational task, representation and algorithms.

3 Proposed Framework

Assuming affect and its change in time is governed by "laws", we build upon our previous work presented in [6] and formally define affect as a continuous time-dependent state vector $\mathbf{a}(t)$, valued in a state-space $\mathcal{A}$: $\mathbf{a}(t) \triangleq \left(v, a, \frac{dv}{dt}, \frac{da}{dt}, \ldots, \frac{d^n v}{dt^n}, \frac{d^n a}{dt^n}\right)^T \in \mathcal{A} \subset \mathbb{R}^{2(n+1)}$, consisting of a level of valence $v \in [-1, 1]$, arousal $a \in [-1, 1]$ and their higher-order time derivatives of order $n \in \mathbb{N}_0$. Assuming the incoming visual observations are only available at discrete time steps t_k, we denote affect using a discrete-time state vector $\mathbf{a}_k$ in stead of the continuous-time state vector $\mathbf{a}(t)$. As such, the mathematical problem of affect estimation can be formulated as:

$$p(\mathbf{a}_k | \mathbf{y}_1, \ldots, \mathbf{y}_k) \tag{1}$$

denoting the inference of the posterior belief of the current affective state $\mathbf{a}_k$ given a stream of visual observations $\{\mathbf{y}_1, \ldots, \mathbf{y}_k\}$ acquired from time-step t_1 up to and including time-step t_k. Key to solving inference problem (1) is the extraction of informative visual observations $\mathbf{y}_k$ from images $\mathbf{I}_k \in \mathcal{I} \subset \mathbb{R}^{H \times W}$. To this end, we propose to use deep *convolutional neural networks* (CNN)s for inducing an observation space $\mathcal{Y} \subset \mathbb{R}^q$ in which the frame-based observations $\mathbf{y}_k$ are valued. Deep CNNs are typically structured as a set of alternating *convolutional* and *pooling* layers followed by *spatial pyramid layers* (SPP) and eventually *fully-connected*

layers. We can therefore formally define deep CNNs as a composition of mappings:

$$\begin{aligned}\mathbf{y}_k &= \mathbf{h}_{\text{CNN}}(\mathbf{x}_k) \\ &\triangleq (\mathbf{h}^{(L)} \circ \mathbf{h}^{(L-1)} \circ \ldots \circ \mathbf{h}^{(2)} \circ \mathbf{h}^{(1)})(\mathbf{x}_k)\end{aligned} \tag{2}$$

where each mapping $\mathbf{h}^{(l)} : \mathcal{X}^{(l)} \rightarrow \mathcal{X}^{(l+1)}, \forall l \in [1, L]$ represents a (trainable) *layer* and the inputs $\mathbf{x}_k$ are canonical face representations extracted from the raw images $\mathbf{I}_k$.

In the specific context of continuous affect estimation from facial expression, training a deep CNN (2) using the widely-used supervised learning algorithms is practically useless due to the lack of reliable one-to-one correspondences between the image frames $\mathbf{I}_k$ and states $\mathbf{a}_k$ [5]. We therefore advocate for a greedy layer-wise unsupervised training strategy of deep CNNs. The remainder of this section gives a detailed description of our three main technical contributions: (i) canonical face representation from raw images, (ii) convex unsupervised representation learning for training deep CNNs and (iii) continuous affect estimation using Bayesian filtering, implemented using neural networks.

3.1 Canonical Face Representation

A fundamental bottleneck in facial expression analysis is the presence of nuisance factors such as head pose, scale and the expressive person's identity. To alleviate this issue, we seek to automatically transform unconstrained images into *canonical face representations* consisting of frontalized faces with standardized scale, 2D shape and uniform background. Face frontalization has recently been shown to substantially boost the performance of various tasks such as face recognition and face verification [43], [44]. We therefore adopt a similar strategy for the task of facial expression analysis and propose a face frontalization algorithm consisting of three processing steps: (i) facial feature point tracking, (ii) neutral (non-expressive) face shape estimation and (iii) piecewise affine texture warping. As facial feature point tracking is not part of our contributions, we only provide a detailed description of neutral face shape estimation and texture warping.

3.1.1 Neutral Face Shape Estimation

A challenging problem in facial expression analysis is how to suppress the information pertaining to the expressive person's identity. Using a shape model that decouples shape deformations due to expressions from shape deformations due to identity is one way to do this [45]. Another way is to subtract the neutral face shape of the person from the expressive shape and use the resulting face for analysis. In some cases the neutral face shape is available as the first frame of a video sequence. However, it often occurs that a video sequence starts with an expressive face or we are given only one expressive face image to analyze. In these cases, one has to estimate the neutral (i.e. expressionless) face shape given the expressive face.

We propose a novel algorithm for directly estimating the neutral shape corresponding to an expressive shape using *partial least squares* (PLS) regression [46]. The intuition behind the algorithm is that, since face shapes are highly structured data objects, they lie on a lower-dimensional

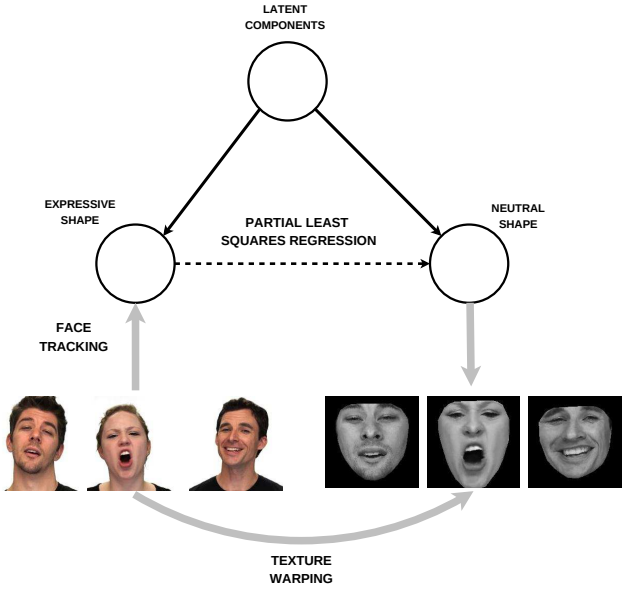


Fig. 1: *Partial least squares* (PLS) regression based face frontalization for two-dimensional canonical face representation.

manifold. As such, our method uses PLS regression for "learning" to map expressive shapes and the corresponding neutral shapes to a shared linear subspace in which they are highly correlated. More formally, let $\{\mathbf{s}_{e,i}, \mathbf{s}_{n,i}\}_{i=1}^{N}$ be a set of available data samples acquired using a facial feature point tracker, where $\mathbf{s}_{e,i} \in \mathbb{R}^p$ represent the expressive shapes and where the point coordinates are collapsed in a p-dimensional vector (predictive variables) and $\mathbf{s}_{n,i} \in \mathbb{R}^p$ denotes the p-dimensional neutral shapes (response variables). Let $\bar{\mathbf{s}}_e$ and $\bar{\mathbf{s}}_n$ denote the sample means for the predictive and response variables respectively:

$$\bar{\mathbf{s}}_e = \frac{1}{N}\sum_{i=1}^{N}\mathbf{s}_{e,i} \quad \text{and} \quad \bar{\mathbf{s}}_n = \frac{1}{N}\sum_{i=1}^{N}\mathbf{s}_{n,i}. \tag{3}$$

Also, let a prime notation denote centered variables, i.e. the subtraction of the sample average: $\mathbf{s}'_{e,i} = \mathbf{s}_{e,i} - \bar{\mathbf{s}}_e$ and $\mathbf{s}'_{n,i} = \mathbf{s}_{n,i} - \bar{\mathbf{s}}_n$. The centered data is collected in two $N \times p$ matrices, $\mathbf{S}_e$ and $\mathbf{S}_n$, as follows:

$$\mathbf{S}_e = \begin{pmatrix} \mathbf{s}'^T_{e,1} \\ \vdots \\ \mathbf{s}'^T_{e,N} \end{pmatrix} \quad \text{and} \quad \mathbf{S}_n = \begin{pmatrix} \mathbf{s}'^T_{n,1} \\ \vdots \\ \mathbf{s}'^T_{n,N} \end{pmatrix}. \tag{4}$$

PLS seeks for a set of M components (latent vectors) that carry out a simultaneous decomposition of $\mathbf{S}_e$ and $\mathbf{S}_n$, with the restriction that these components explain the maximum covariance between both sets. SIMPLS [47], a fast iterative PLS variant, decomposes the data matrices as (for this description, we draw partially on [48])

$$\mathbf{S}_e = \mathbf{T}\mathbf{A}_e^T + \mathbf{R}_e, \tag{5}$$

$$\mathbf{S}_n = \mathbf{T}\mathbf{A}_n^T + \mathbf{R}_n, \tag{6}$$

where $\mathbf{T}$ is an orthogonal $(N \times M)$ matrix of latent vectors, $\mathbf{T}\mathbf{T}^T = \mathbf{I}_N$, the $(p \times M)$ matrices $\mathbf{A}_e$ and $\mathbf{A}_n$ represent matrices of loadings, and $\mathbf{R}_e$ and $\mathbf{R}_n$ are $N \times p$ matrices of residuals. For computing $\mathbf{T}$, SIMPLS constructs a linear transformation of $\mathbf{S}_e$, $\mathbf{T} = \mathbf{S}_e\mathbf{W}_e$, where $\mathbf{W}_e$ is a $p \times M$

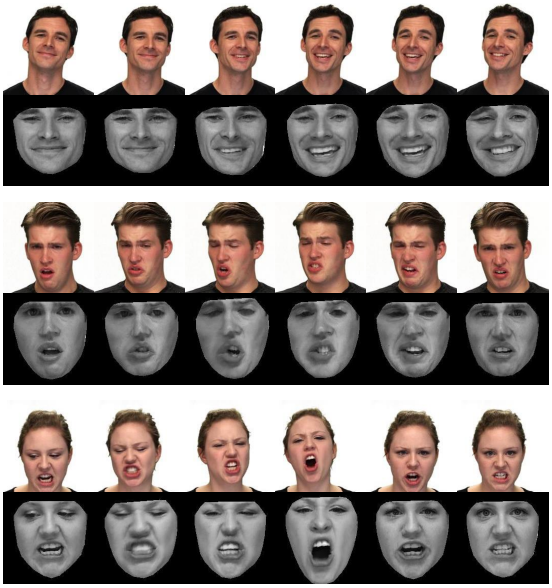


Fig. 2: Visualization of our proposed expressive face frontalization method on testing faces from the RAVDESS dataset.

weight matrix. To this end, SIMPLS finds at each iteration two sets of weights $\mathbf{w}_e$ and $\mathbf{w}_n$ (columns of $\mathbf{W}_e$ and $\mathbf{W}_n$ respectively) in order to create a linear combination of the columns of $\mathbf{S}_e$ and $\mathbf{S}_n$ such that their covariance is maximized [49]. The latent components $\mathbf{T}$ are then used for prediction in place of the original variables, that is $\mathbf{A}_n^T$, which can be obtained from $\mathbf{T}$ as the least squares solution of (6):

$$\mathbf{A}_n^T = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{S}_n \tag{7}$$

Then, substituting $\mathbf{T}$ in (6) yields $\mathbf{S}_n = \mathbf{S}_e\mathbf{W}_e\mathbf{A}_n^T + \mathbf{R}_n$, hence, the matrix $\mathbf{W}$ of regression coefficients for the model $\mathbf{S}_n = \mathbf{S}_e\mathbf{W} + \mathbf{R}_n$ is given by

$$\mathbf{W} = \mathbf{W}_e\mathbf{A}_n^T \tag{8}$$

For a non-centered test sample $\mathbf{s}_e$ (expressive shape), the predicted response $\mathbf{s}_n$ (neutral shape) is computed as follows:

$$\mathbf{s}_n = \mathbf{W}^T\mathbf{s}_e + (\bar{\mathbf{s}}_n - \mathbf{W}^T\bar{\mathbf{s}}_e). \tag{9}$$

Figure 1 depicts our proposed PLS-based face frontalization algorithm. In the following paragraph we briefly explain how the resulting neutral face shape estimation is combined with the input image for synthesizing a frontal view of the face.

3.1.2 Piecewise Affine Texture Warping

Following the notation introduced above, let $\mathbf{s}_e^* \in \mathbb{R}^p$ and $\mathbf{s}_n^* \in \mathbb{R}^p$ denote the normalized (i.e. horizontally straightened) expressive face shape and the normalized neutral face shape estimation, respectively. In order to only retain the expression information, we subtract the neutral shape from the expressive shape:

$$\Delta\mathbf{e}^* = \mathbf{s}_e^* - \mathbf{s}_n^*, \tag{10}$$

Given this normalized expressive face shape increment (deformation) $\Delta\mathbf{e}^*$ and the associated appearance image $\mathbf{I}$, we are interested to build an appearance image for which the identity information is "removed" from the associated shape and only the expression information remains. To this end, we construct the average normalized neutral shape $\bar{\mathbf{s}}_n^* = \sum_{i=1}^N \mathbf{s}_{n,i}^*$. We then transfer the deformation information $\Delta\mathbf{e}^*$ to $\bar{\mathbf{s}}_n^*$ and obtain a canonical face shape $\tilde{\mathbf{s}}_e$:

$$\tilde{\mathbf{s}}_e = \bar{\mathbf{s}}_n^* + \Delta\mathbf{e}^*. \tag{11}$$

We finally deform, through a *piecewise affine texture warping* operation, the appearance image $\mathbf{I}$ with the original face shape $\mathbf{s}_e$ so that it matches the canonical face shape $\tilde{\mathbf{s}}_e$ (11). This transformation entails that the shapes are triangulated in the same manner and each triangle patch appearance in the source image is affinely warped so that it aligns with the target triangle [50]. As a result of the transformation, we obtain a *canonical face representation* $\mathbf{x} \in \mathbb{R}^{d\times d}$ in the form of a gray-scale image consisting of a centered and normalized face shape and texture, with an uniform (black) background. We believe that such a representation yields a better starting point for the subsequent deep learning based facial expression representation and analysis tasks.

In the remainder of this paper, we assume the PLS-based neutral face shape estimator is trained using the *extended Cohn-Kanade* (CK+) dataset [14]. The choice of CK+ is justified by its reliable neutral face shapes $\{\mathbf{s}_{n,i}\}_{i=1}^N$ present in the first frame of each sequence. Although trained on the relatively simple CK+ dataset, figure 2 shows the generalization capability of our PLS method on more challenging and unseen images from the *Ryerson audio-visual database of emotional speech and song* (RAVDESS) dataset [51].

3.2 Deep Unsupervised Representation Learning

Generally, representation learning aims at learning to identify and disentangle the underlying explanatory factors hidden in the observed data, while taking into account general priors such as: invariance, smoothness, a hierarchical organization of explanatory factors, shared factors across tasks, temporal and spatial coherence, manifolds, sparsity [52]. In the particular case of facial expression analysis, image variabilities such as camera viewpoint, lighting condition and pose complicate image representation. The recent empirical success of deep *convolutional neural networks* (CNN)s suggests that their compositional and hierarchical structure induces increasingly invariant data representations by progressively flattening and separating the manifold-shape of the data [53]. Based on our previous work presented in [12], [54], we present a novel algorithmic framework for learning to extract data representations from large-scale image data, based on deep CNNs as computational models.

3.2.1 Reconstruction Contractive Auto-Encoders

The unsupervised representation learning problem can be posed as the optimization of the parameters $\boldsymbol{\theta} = (\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(K)})$ of a CNN given an unlabeled dataset $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N$, subject to discovering the local properties of the data-generating distribution $p(\mathbf{x})$. There exist two general frameworks for learning the parameters of a CNN while characterizing the data-generating distribution, one is a probabilistic framework and the other is a deterministic framework. The probabilistic framework aims at learning

the data-generating distribution directly from the unlabeled data-set by proposing a parametrized joint distribution $p(\mathbf{x}, \mathbf{h}; \boldsymbol{\theta})$ over the observed data $\mathbf{x} \in \mathcal{X}$ and hidden representations $\mathbf{h} = \mathbf{h}(\mathbf{x}) \in \mathcal{H}$. Then, by maximizing the data likelihood, one can obtain the optimal parameters $\boldsymbol{\theta}$ of the CNN mapping $\mathbf{h} : \mathcal{X} \rightarrow \mathcal{H}$. The most popular model for the parametrized joint distribution $p(\mathbf{x}, \mathbf{h}; \boldsymbol{\theta})$ is the *convolutional restricted Boltzmann machine* (cRBM) [55]. Despite its success, training a CNN using such a model is known to be computationally intractable. In contrast to RBMs, deterministic algorithms have been developed, the most successful being the *auto-encoder* (AE) [52].

AEs directly learn the mapping $\mathbf{h} : \mathcal{X} \rightarrow \mathcal{H}$ by optimizing its parameters such that the input data $\mathbf{x}$ can be reconstructed as accurate as possible while satisfying some constraints. Some recent results have shown that when AEs are trained using a reconstruction contraction criterion, they learn local properties of the data-generating distribution [13], which is what we aim for. Carefully regularized AEs manage to learn the shape of the high-density regions of the data-generating distribution. There exist many variants for regularizing AEs, e.g. *sparse auto-encoders* (SAE)s, *denoising auto-encoders* (DAE)s, *contractive auto-encoders* (CAE)s, *higher-order contractive auto-encoders* (HCAE)s and *reconstruction contractive auto-encoders* (RCAE)s [56]. Among them, the RCAE scheme has been proven to yield representations that capture the high-density regions of the data-generating distribution [13]. Given a general *non-parametric* reconstruction function $\mathbf{r} : \mathcal{X} \rightarrow \mathcal{X}$ such that $\mathbf{r}(\mathbf{x}) = \mathbf{x}$, an RCAE aims at minimizing the following expected loss:

$$\mathcal{L}_\lambda(\mathbf{r}) = \int_{\mathcal{X}} p(\mathbf{x}) \left[\|\mathbf{r}(\mathbf{x}) - \mathbf{x}\|_2^2 + \lambda \left\| \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} \right\|_F^2 \right] d\mathbf{x} \quad (12)$$

where $\| \bullet \|_F$ is the Frobenius norm. The central result is that any minimizer $\mathbf{r}^*_\lambda(\mathbf{x})$ of the expected loss function defined by (12) allows us to obtain an estimator of the score (first derivative of the log-density) $\frac{\partial \log p(\mathbf{x})}{\partial \mathbf{x}}$, i.e., the direction in which density is increasing the most, which also corresponds to the local mean. It also estimates the Hessian (second derivative of the log-density) $\frac{\partial^2 \log p(\mathbf{x})}{\partial \mathbf{x}^2}$. The exact expression of this claim is formulated in [13].

The theoretical motivation for RCAEs has been shown for a large class of reconstruction (encoding-decoding) functions minimizing the empirical loss $\mathcal{L}_\lambda(\mathbf{r})$ and regularization parameter λ approaching 0 asymptotically [13]. The practical problem of representation learning therefore reduces to the design of efficient algorithms that find the optimal parameters minimizing the empirical loss (12). In the remaining of this section we present our proposed algorithmic framework for solving this minimization problem in the case of convolutional and fully-connected layers.

3.2.2 Training Convolutional Layers

Solving the RCAE objective (12) is often difficult due to the underlying non-convexity. The predominant methodology for solving such problem is the widely used *stochastic gradient descent* (SGD) algorithm [57]. Despite its ease of implementation, SGD is known to be very difficult to tune, parallelize or distribute. For this reason, our primary goal is to transform the RCAE objective into a convex optimization problem, hence *convex unsupervised representation learning* (CURL). We present a novel algorithm for efficiently solving the RCAE problem. The rationale behind our algorithm is two-fold: (i) *random convexification*, i.e. fixing the non-linear encoding filters randomly and only learning the (untied) linear decoding filters, (ii) *spectral minimization*, i.e. learning the decoding filters in the frequency domain. As a direct consequence, the main computational advantages are: (i) very few hyper-parameters to tune, and (ii) fast and guaranteed convergence. More formally, we consider a convolutional reconstruction function with K filters and input space $\mathcal{X} \subset \mathbb{R}^{H \times W \times C}$, i.e. the space of C-channel $H \times W$ images $\mathbf{x} = \left(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(C)}\right)$. We then define the following per-channel zero-bias convolutional reconstruction function:

$$\begin{aligned} \mathbf{r}^{(c)}(\mathbf{x}; \boldsymbol{\theta}) &\triangleq \underbrace{\sum_{k=1}^{K} \mathbf{w}^{(k,c)}}_{\text{linear decoding}} * \underbrace{\mathbf{g}\Big(\sum_{l=1}^{C} \mathbf{a}^{(k,l)} * \mathbf{x}^{(l)} + \mathbf{b}^{(k)}\Big)}_{\text{random nonlinear encoding}} \\ &= \sum_{k=1}^{K} \mathbf{w}^{(k,c)} * \mathbf{h}^{(k)} \end{aligned} \quad (13)$$

with $\mathbf{g}(\cdot)$ denoting the element-wise application of an activation function $g : \mathbb{R} \rightarrow \mathbb{R}$. The model parameters $\boldsymbol{\theta}$ consist of encoding filters $\mathbf{a}^{(k,l)}$, encoding biases $\mathbf{b}^{(k)}$ and decoding filters $\mathbf{w}^{(k,c)}$. We further propose to sample the (non-linear) encoding parameters $\mathbf{a}^{(k,l)}$ and $\mathbf{b}^{(k)}$ from pre-determined density functions $p(\mathbf{a})$ and $p(\mathbf{b})$ respectively, and keep them fixed while learning the untied (linear) decoding parameters $\mathbf{w}^{(k,c)}$. When substituting the convolutional reconstruction function (13) into the RCAE objective (12), applied to each input channel, we obtain an empirical objective of the form $\mathcal{L}_\lambda = \sum_{c=1}^{C} \mathcal{L}_\lambda^{(c)}$, where the per-channel empirical objectives are defined as:

$$\begin{aligned} \mathcal{L}_\lambda^{(c)} = \frac{1}{N} \sum_{n=1}^{N} &\left\| \sum_{k=1}^{K} \mathbf{w}^{(k,c)} * \mathbf{h}_n^{(k)} - \mathbf{x}_n^{(c)} \right\|_F^2 \\ &+ \lambda \left\| \sum_{k=1}^{K} \mathbf{w}^{(k,c)} * \frac{\partial \mathbf{h}_n^{(k)}}{\partial \mathbf{x}_n^{(c)}} \right\|_F^2 \end{aligned} \quad (14)$$

with $\mathbf{h}_n$ denoting the randomized encoding of the n-th training sample $\mathbf{x}_n$. In this case, minimizing the RCAE objective reduces to solving a linear least-squares problem, yielding a good speed-accuracy trade-off. Although such a random convexification scheme has recently been proposed in the context of multi-layer perceptrons [58], [59] and more general models [60], it has, to the best of our knowledge, never been applied to the convolutional case.

To train the convexified RCAE (14), we adopt a complex-valued spectral (re)parametrization and define the complex-valued decoding parameters associated to $\mathbf{w}^{(k,c)}$ as $\mathbf{W}^{(k,c)}$, with $\mathbf{W}^{(k,c)} = \mathcal{F}\{\mathbf{w}^{(k,c)}\} \in \mathbb{C}^{H \times W}$ being the *discrete Fourier transform* (DFT). Then, using Parseval's theorem in conjunction with the convolution theorem [61], we transform the convolution operation into an element-wise multiplication,

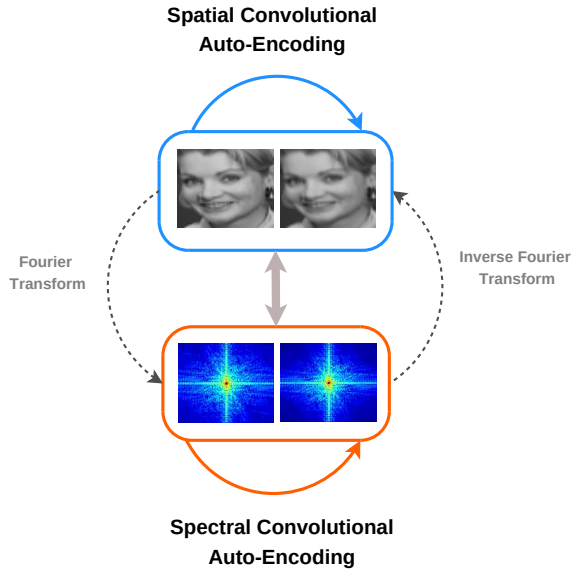


Fig. 3: Illustration of equivalence between spatial convolutional auto-encoding and spectral convolutional auto-encoding. A good spectral reconstruction yields a good spatial reconstruction and vice-versa.

yielding the following per-channel spectral RCAE objective:

$$\mathcal{L}_{\lambda}^{(c)} \propto \frac{1}{N}\sum_{n=1}^{N}\left\|\sum_{k=1}^{K}\mathbf{W}^{(k,c)}\odot\mathbf{H}_n^{(k)}-\mathbf{X}_n^{(c)}\right\|_F^2 + \lambda\left\|\sum_{k=1}^{K}\mathbf{W}^{(k,c)}\odot\mathbf{D}_n^{(k,c)}\right\|_F^2 \tag{15}$$

where $\odot$ denotes the Hadamard (element-wise) product [62], $\mathbf{H}_n^{(k)} = \mathcal{F}\{\mathbf{h}_n^{(k)}\}$, $\mathbf{X}_n^{(c)} = \mathcal{F}\{\mathbf{x}_n^{(c)}\}$, $\mathbf{D}_n^{(k,c)} = \mathbf{G}_n^{(k)} \odot \mathcal{F}\{\mathbf{a}^{(k,c)}\}$ with $\mathbf{G}_n^{(k)} = \mathcal{F}\left\{\frac{\partial g(\mathbf{v})}{\partial \mathbf{v}}\Big|_{\mathbf{v}=\mathbf{v}_n^{(k)}}\right\}$ and $\mathbf{v}_n^{(k)} = \sum_{l=1}^{C}\mathbf{a}^{(k,l)} * \mathbf{x}_n^{(l)} + \mathbf{b}^{(k)}$. As a direct consequence, minimizing the spectral RCAE objective (15) reduces to solving $H.W$ independent K-dimensional complex-valued regularized linear least-squares problems. We propose to solve each of the independent K-dimensional problems using the *Gauss-Seidel* (GS) algorithm [62], [63]. We refer to [12] for a derivation fo the GS iterations and only provide its final form:

$$\mathbf{W}_t^{(k,c)} = \left(\mathbf{B}^{(k)(c)} - \sum_{l=1}^{k-1}\mathbf{A}^{(k,l)(c)}\odot\mathbf{W}_t^{(l,c)} - \sum_{l=k+1}^{K}\mathbf{A}^{(k,l)(c)}\odot\mathbf{W}_{t-1}^{(l,c)}\right)\oslash\mathbf{A}^{(k,k)(c)} \tag{16}$$

where $\oslash$ denotes the element-wise division. Thanks to this nice convergence properties of the GS algorithm, in practice, the iterations are stopped when the approximate relative error $\epsilon_t^{(k,c)} = \|\mathbf{W}_t^{(k,c)} - \mathbf{W}_{t-1}^{(k,c)}\|_F / \|\mathbf{W}_t^{(k,c)}\|_F$ is less than a pre-determined tolerance τ for each filter or when t reaches a pre-defined maximum number of iterations T. Figure 3 illustrates the duality between convolutional auto-encoding in the spatial domain and the spectral domain, i.e. a good spectral reconstruction yields a good spatial reconstruction and vice-versa. This clearly highlights the advantages of random convexification and spectral minimization using the GS algorithm.

Once the decoding filters are learned in the frequency domain, they are transformed back to the spatial domain using the inverse DFT: $\hat{\mathbf{w}}^{(k,c)} = \mathcal{F}^{-1}\{\mathbf{W}_t^{(k,c)}\}$. At inference stage, the resulting per-channel output of a zero-bias convolutional layer $\mathbf{h}^{(l)} : \mathcal{X}^{(l)} \rightarrow \mathcal{X}^{(l+1)}$ is defined as:

$$\mathbf{x}^{(k)(l+1)} \triangleq \mathbf{g}\left(\sum_{c=1}^{C}\hat{\mathbf{w}}^{(k,c)T} * \mathbf{x}^{(c)(l)}\right) \tag{17}$$

In the following, we apply the same convexification strategy (i.e. CURL) to the RCAE problem for fully-connected layers. Different from the convolutional layers, in the case of fully connected layers we do not solve the problem in the frequency domain but solve it as a conventional (spatial) matrix-vector linear least-squares problem.

3.2.3 Training Fully-Connected Layers

Similar to the convolutional layers, we propose the following zero-bias *fully-connected auto-encoder* (FAE) as reconstruction function:

$$\mathbf{r}(\mathbf{x};\boldsymbol{\theta}) \triangleq \underbrace{\mathbf{W}}_{\text{decoding}}\underbrace{\boldsymbol{g}(\mathbf{A}\mathbf{x}+\mathbf{b})}_{\text{encoding}} = \mathbf{W}\mathbf{h} \tag{18}$$

with model parameter $\boldsymbol{\theta}$ consisting of an $h \times d$ encoding matrix $\mathbf{A}$, $h \times 1$ encoding bias $\mathbf{b}$ and $d \times h$ decoding matrix $\mathbf{W}$. In the case of fully-connected layer, the input space is considered to be $\mathcal{X} \subset \mathbb{R}^d$, i.e. the space of d-dimensional vectors. By substituting the FAE reconstruction function (18) in the RCAE objective (12) and fixing the (non-linear) encoding parameters $\{\mathbf{A}, \mathbf{b}\}$ randomly, we can fit the (linear) decoding parameter $\boldsymbol{\theta} = \mathbf{W}$ optimally using a *convex minimization* strategy for the following objective:

$$\mathcal{L}_{\lambda}(\mathbf{W}) = \frac{1}{N}\sum_{n=1}^{N}\|\mathbf{W}\mathbf{h}_n - \mathbf{x}_n\|_F^2 + \lambda\|\mathbf{W}\mathbf{D}_n\|_F^2 \tag{19}$$

where $\mathbf{D}_n = \mathbf{G}_n\mathbf{A}$ with $\mathbf{G}_n = \frac{\partial \mathbf{g}}{\partial \mathbf{v}}|_{\mathbf{v}=\mathbf{v}_n}$ and $\mathbf{v}_n = \mathbf{A}\mathbf{x}_n + \mathbf{b}$. Minimizing this objective reduces to solving a linear least-squares minimization problem with Tikhonov regularization, which has the following closed-from solution:

$$\hat{\mathbf{W}}^T = \left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{D}^T\mathbf{D}\right)^{-1}\left(\mathbf{H}^T\mathbf{X}\right) \tag{20}$$

In practice, we solve the linear least-squares problem (19) incrementally in order to handle datasets of arbitrary size using relatively few resources. At inference stage, the resulting output of a zero-bias fully-connected layer $\mathbf{h}^{(l)} : \mathcal{X}^{(l)} \rightarrow \mathcal{X}^{(l+1)}$ is defined as:

$$\mathbf{x}^{(l+1)} \triangleq \mathbf{g}(\hat{\mathbf{W}}^T\mathbf{x}^{(l)}) \tag{21}$$

In summary, we present CURL, a novel reconstruction contractive auto-encoder (RCAE) based algorithmic framework for training CNNs in an unsupervised greedy layer-wise manner. Given a dataset of images, the algorithm trains one CNN layer at the time, according to its type (convolutional or fully-connected). In the context of facial expression analysis, CURL provides a means for training CNNs on top

of the canonical face representations presented in section 3.1. In the following, we present how the resulting CNN representations are used for continuous affect estimation.

3.3 Deep Extended Kalman Filtering

Key to solving the Bayesian filtering problem (1) is to characterize the *state transition density* $p(\mathbf{a}_k|\mathbf{a}_{k-1})$ describing the dynamics of the affective state and the *observation density* $p(\mathbf{y}_k|\mathbf{a}_k)$ describing the likelihood of an observation given the affective state. This probabilistic perspective is ideally suited for a Bayesian treatment, which provides a rigorous and general framework for probabilistic reasoning over time [64]. Assuming the affective states follow a first-order Markov process and that the observations are independent of the given state, the general affective state estimation problem can be derived as computing (at each time step t_k) a *belief* (posterior distribution) of the affective state $\mathbf{a}_k$ given all observations up to and including time step t_k [6]:

$$\begin{aligned} p(\mathbf{a}_k|\mathbf{y}_{1:k}) &\propto p(\mathbf{y}_k|\mathbf{a}_k)p(\mathbf{a}_k|\mathbf{y}_{1:k-1}) \\ &= p(\mathbf{y}_k|\mathbf{a}_k)\int_{\mathcal{A}} p(\mathbf{a}_k|\mathbf{a}_{k-1})p(\mathbf{a}_{k-1}|\mathbf{y}_{1:k-1})d\mathbf{a}_{k-1} \end{aligned} \tag{22}$$

where the so-called Chapman-Kolmogorov integral is known to be computationally intractable in most cases [7]. In the particular case of linear and Gaussian transition and observation models, this integral becomes tractable and the filtering problem then has a closed form solution, i.e. the much celebrated *Kalman filter* (KF) [65].

Despite being successfully used in our previous work [6], the use of a Kalman filter is often limited by the ubiquitous non-linearity and non-Gaussianity of the physical world and numerous efforts have therefore been devoted to the non-linear filtering problem in the Kalman filtering framework [7]. In this paper, we follow the same idea and extend the classical linear Kalman filtering algorithm to its non-linear case, the so-called *extended Kalman filtering* (EKF), using *neural network* (NN) transition and observation models, hence *deep extended Kalman filtering* (DEKF). The advantage of using a non-linear approach is the additional capacity of modeling complex processes with a relatively small computation overhead, compared to the linear KF. We therefore define the following non-linear transition and observation densities:

$$\begin{aligned} p(\mathbf{a}_k|\mathbf{a}_{k-1}) &= \mathcal{N}(\mathbf{f}_{\text{NN}}(\mathbf{a}_{k-1}), \mathbf{Q}) \\ &= \mathcal{N}(\mathbf{W}_f\boldsymbol{\phi}(\mathbf{A}_f\mathbf{a}_{k-1} + \mathbf{b}_f), \mathbf{Q}) \\ p(\mathbf{y}_k|\mathbf{a}_k) &= \mathcal{N}(\mathbf{g}_{\text{NN}}(\mathbf{a}_k), \mathbf{R}) \\ &= \mathcal{N}(\mathbf{W}_g\boldsymbol{\phi}(\mathbf{A}_g\mathbf{a}_k + \mathbf{b}_g), \mathbf{R}) \end{aligned} \tag{23}$$

where the transition model $\mathbf{f}_{\text{NN}}$ and observation model $\mathbf{g}_{\text{NN}}$ are *single hidden layer neural networks* (SHLNN)s parametrized by the weights and biases $\{\mathbf{W}_f, \mathbf{A}_f, \mathbf{b}_f\}$ and $\{\mathbf{W}_g, \mathbf{A}_g, \mathbf{b}_g\}$ respectively, and a non-linear activation $\phi : \mathbb{R} \rightarrow \mathbb{R}$. The matrices $\mathbf{Q}$ and $\mathbf{R}$ are the transition noise covariance and observation noise covariance respectively.

Algorithm 1: Deep Extended Kalman Filtering

input : Transition model $\mathbf{f}_{\text{NN}}$; Observation model $\mathbf{g}_{\text{NN}}$; Deep CNN model $\mathbf{h}_{\text{CNN}}$; Initial state prior $\{\hat{\mathbf{a}}_0, \hat{\mathbf{P}}_0\}$; Visual stimuli $\{\mathbf{x}_k\}_{k=1}^K$

output: Sequence of estimated mean and covariance $\{\hat{\mathbf{a}}_k, \hat{\mathbf{P}}_k\}_{k=1}^K$

for each time step t_k **do**

$\hat{\mathbf{a}}_{k|k-1} = \mathbf{f}_{\text{NN}}(\hat{\mathbf{a}}_{k-1})$ //state prediction

$\hat{\mathbf{F}} = \partial_{\mathbf{a}}\mathbf{f}_{\text{NN}}|_{\mathbf{a}=\hat{\mathbf{a}}_{k-1}}$

$\hat{\mathbf{P}}_{k|k-1} = \hat{\mathbf{F}}\hat{\mathbf{P}}_{k-1}\hat{\mathbf{F}}^T + \hat{\mathbf{Q}}$ //covariance prediction

$\hat{\mathbf{G}} = \partial_{\mathbf{a}}\mathbf{g}_{\text{NN}}|_{\mathbf{a}=\hat{\mathbf{a}}_{k|k-1}}$

$\mathbf{K} = \hat{\mathbf{P}}_{k|k-1}\hat{\mathbf{G}}^T[\hat{\mathbf{G}}\hat{\mathbf{P}}_{k|k-1}\hat{\mathbf{G}}^T + \hat{\mathbf{R}}]^{-1}$ //Kalman gain

$\mathbf{y}_k = \mathbf{h}_{\text{CNN}}(\mathbf{x}_k)$ //deep CNN observation

$\hat{\mathbf{a}}_k = \hat{\mathbf{a}}_{k|k-1} + \mathbf{K}[\mathbf{y}_k - \mathbf{g}_{\text{NN}}(\hat{\mathbf{a}}_{k|k-1})]$ //state update

$\hat{\mathbf{P}}_k = \hat{\mathbf{P}}_{k|k-1} - \mathbf{K}\hat{\mathbf{G}}\hat{\mathbf{P}}_{k|k-1}$ //covariance update

end

Based on the first-order Taylor series expansion, the rationale behind EKF is the linearization of the transition and observation models around the estimated trajectory:

$$\begin{aligned} \hat{\mathbf{F}} &\triangleq \left.\frac{\partial \mathbf{f}_{\text{NN}}}{\partial \mathbf{a}}\right|_{\mathbf{a}=\hat{\mathbf{a}}_{k-1}} \\ &= \mathbf{W}_f \text{diag}\left(\boldsymbol{\phi}'(\mathbf{A}_f\mathbf{a}_{k-1} + \mathbf{b}_f)\right)\mathbf{A}_f \\ \hat{\mathbf{G}} &\triangleq \left.\frac{\partial \mathbf{g}_{\text{NN}}}{\partial \mathbf{a}}\right|_{\mathbf{a}=\hat{\mathbf{a}}_{k|k-1}} \\ &= \mathbf{W}_g \text{diag}\left(\boldsymbol{\phi}'(\mathbf{A}_g\mathbf{a}_k + \mathbf{b}_g)\right)\mathbf{A}_g \end{aligned} \tag{24}$$

where $\boldsymbol{\phi}'$ denotes the element-wise application of the first-order derivative function of the non-linear activation $\phi : \mathbb{R} \rightarrow \mathbb{R}$. As a result of the linearization (24), the filtering problem becomes linear and Gaussian and, as a direct consequence, the conventional (linear and Gaussian) KF algorithm is further employed for finding the optimal state estimation. Moreover, the linearity and Gaussianity of the transition and observation models guaranties that an initial Gaussian density of the state vector $p(\mathbf{a}_0)$ at time-step t_0 remains Gaussian for all time-steps t_k. The EKF is therefore entirely characterized by its mean and covariance estimates $\{\hat{\mathbf{a}}_k, \hat{\mathbf{P}}_k\}$. The DEKF neural network weights and biases $\{\mathbf{W}_f, \mathbf{W}_g, \mathbf{A}_f, \mathbf{A}_g, \mathbf{b}_f, \mathbf{b}_g\}$ are estimated using the *extreme learning machine* (ELM) strategy (i.e. random input-to-hidden weights and learned hidden-to-output weights [66]) and by creating a training dataset $\mathcal{D} = \{\mathbf{a}_i, \mathbf{y}_i\}_{i=1}^N$ consisting of latent states $\mathbf{a}_i$, i.e. valence, arousal and higher-order derivatives, and informative observations $\mathbf{y}_i$. Similar to method presented in [6], the transition and observation covariance matrices $\mathbf{Q}$ and $\mathbf{R}$ are estimated using a regularized linear least squares approach on the residuals.

Algorithm 1 summarizes the resulting DEKF. The practical benefit is that the DEKF algorithm can monitor the latent state and its uncertainty while acquiring visual stimuli and keeping the memory resources fixed without too much computational overhead compared to its linear counterpart.

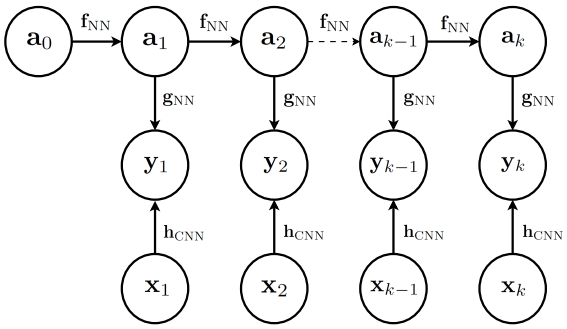


Fig. 4: Graphical model of a *deep extended Kalman filter* (DEKF). A DEKF performs *abductive* inference, i.e. inferring the most probable hidden causes $\mathbf{a}_{1:k}$ from the observable effects $\mathbf{y}_{1:k}$, using a recursive Bayesian approach. The observable effects $\mathbf{y}_{1:k}$ are inferred from the visual stimuli $\mathbf{x}_{1:k}$ using a deep CNN.

Figure 4 depicts the general concept of using NNs for modeling transition and observation functions and inferring the latent states from deep CNN observations using a DEKF approach. It is interesting to observe the conceptual similarity between DEKF and other well-known neural network architectures such as *recursive neural networks* and, more specifically, *recurrent neural networks* (RNN)s. Both models are *directed acyclic graphs* (DAG)s with edges modeled as feed-forward neural networks. Both models have a linear temporal structure and contain partially observable layers, i.e. an observable and a non-observable layer, stacked on top of each-other. A toy experiment for benchmarking our DEKF algorithm is presented in the supplemental material.

4 Empirical Evaluation

In the following, we empirically evaluate the performance of our full pipeline, consisting of three components: (i) *canonical face representation* (CFR), (ii) *convex unsupervised representation learning* (CURL), and (iii) *deep extended Kalman filtering* (DEKF). The performance of the resulting CFR-CURL-DEKF algorithmic framework is empirically validated by means of quantitative experimental results on publicly available benchmark datasets for facial expression recognition and continuous affect estimation. To evaluate the added value of the CFR and CURL components in terms of facial expression representation, we use an *extreme learning machine* (ELM) as facial expression classifier, hence CFR-ELM and CFR-CURL-ELM. For the CFR-ELM model, we simply vectorize the CFR images and use them as input to an ELM.

4.1 Facial Expression Recognition

In this experiment, we assess the performance of our proposed CURL algorithmic framework on the task of facial expression recognition. Although this task is not the main focus of this paper, it is essential for benchmarking our representation learning approach and for getting insight for the subsequent task of continuous affect estimation from facial expressions. As benchmark dataset, we use the *extended Cohn-Kanade* (CK+) dataset [14]. The CK+ dataset consists of 529 videos from 123 subjects, all of them annotated with six expression labels and 327 of them annotated with eight expression labels. On top of the canonical face representations, we apply our CURL algorithm for greedy layer-wise unsupervised training of a deep CNN, as illustrated in figure 6. To achieve this, we stacked a convolutional and a fully-connected layer. More precisely, the resulting CNN architecture consists of five layers: one convolutional layer with 25 filters of size 5×5, a 3×3 max-pooling layer with stride equal 2, a 21-bin spatial pyramid pooling layer and a 2000-dimensional fully-connected layer, followed by a 300-dimensional extreme learning machine. In this configuration, the input-to-hidden layer of the ELM is considered as a fully-connected layer, hence the resulting CNN architecture is referred to as 25c5-2000f-300f. The SPP layer has a pyramid structure of $\{4 \times 4, 2 \times 2, 1 \times 1\}$ spatial bins (21 in total) and transforms the max-pooling layer into a 525-dimensional vector.

As evaluation protocol, we follow a similar approach as other works and we extract the last three frames of each video to compose an image-based CK+ dataset. The total number of images is then split into 10 folds and the subjects are divided into 10 groups. All the images have been preprocessed using a whitening transform. We compare our different models with traditional approaches such as CSPL [67], AdaGabor [68], LBP-SVM [69] and STM [70] on the one hand, and deep-learning based approaches such as 3DCNN-DAP [71], BDBN [30], DTAGN [72], Inception [73], LOMo [74], PPDN [32], Zero-bias CNN [31], FN2EN [34], AURF [26] and AUDN [75] on the other hand. The results are as reported by their respective references. Tables 1 and 2 summarize the obtained classification results.

Table 1 reports comparative facial expression recognition results for the following six emotion classes: "Anger", "Disgust", "Fear", "Happiness", "Sadness" and "Surprise". We observe that our CFR-ELM model reaches an accuracy of 93.6% and hence outperforms or is in par with the traditional methods. Our CFR-CURL-ELM algorithm allows us to gain almost 4% in accuracy w.r.t. CFR-ELM and outperforms almost all the state-of-the-art deep learning approaches (97.4%), except the Zero-bias CNN (98.3%) and FN2EN (98.6%). Note that all the deep learning approaches extensively use data augmentation and supervised fine-tuning, while our approach is entirely unsupervised and uses a very limited amount of data augmentation. When augmenting the data using an horizontal flipping operation, we obtain an classification accuracy of 98.2%. A similar observation can be made for table 2, reporting the classification results for eight emotion classes, i.e. the basic six emotions augmented with "Neutral" and "Contempt". In this case, our baseline contribution CFR-ELM surprisingly outperforms VGG (trained from scratch), zero-bias CNN (without data augmentation) and is in par with VGG (fine-tuned). The combination of CFR with CURL allows us to gain almost 4% w.r.t. the baseline CFR-ELM and outperform AURF. When we further augment the data, our method is in par with the AUDN approach. Note that our approach does not use any supervision, when training the CNN architecture. This is also the reason why, approaches such as Zero-bias CNN (fine-tuned and augmented) and FN2EN outperform our approach.

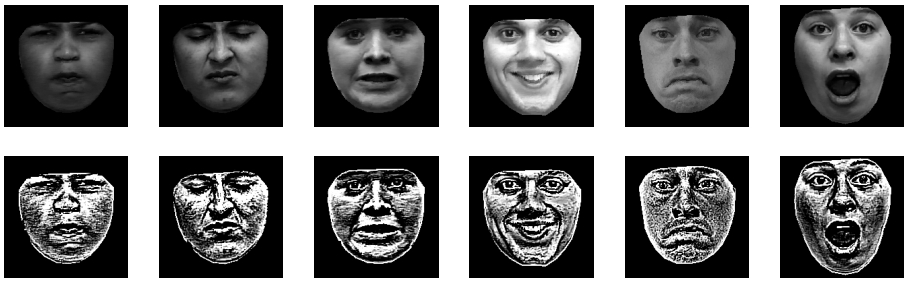


Fig. 5: Example canonical face representations (top) from the CK+ dataset and their whitened counterparts (bottom). From left to right, the expressed emotions are anger, disgust, fear, happiness, sadness and surprise.

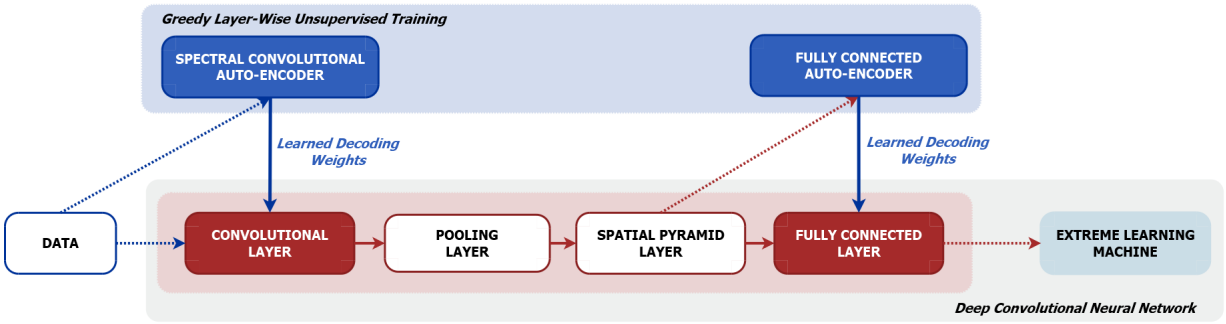


Fig. 6: Deep learning pipeline for facial expression recognition using a given CNN architecture (25c5-2000f-300f). The learning pipeline consists of two phases: (i) greedy layer-wise unsupervised training using our proposed CURL algorithmic framework and (ii) supervised training of an ELM classifier. As input data to the pipeline, we use whitened canonical face representations.

TABLE 1: Comparative facial expression recognition results on the CK+ dataset using a 25c5-2000f-300f CNN architecture. Bold denotes our proposed algorithms, (*) denotes supervised fine-tuning and (†) denotes data augmentation.

Algorithm	Accuracy on 6 classes
CSPL [67]	89.9%
3DCNN-DAP* [71]	92.4%
Inception* [73]	93.2%
AdaGabor [68]	93.3%
CFR-ELM	**93.6%**
STM-ExpLet [70]	94.2%
LBP-SVM [69]	95.1%
LOMo* [74]	95.1%
BDBN* [30]	96.7%
DTAGN* [72]	97.3%
PPDN* [32]	97.3%
CFR-CURL-ELM	**97.4%**
CFR-CURL-ELM†	**98.2%**
Zero-bias CNN*† [31]	98.3%
FN2EN*† [34]	98.6%

TABLE 2: Comparative facial expression recognition results on the CK+ dataset using a 25c5-2000f-300f CNN architecture. Bold denotes our proposed algorithms, (*) denotes supervised fine-tuning and (†) denotes data augmentation.

Algorithm	Accuracy on 8 classes
Zero-bias CNN* [31]	85.6%
VGG from Scratch* [34]	88.7%
CFR-ELM	**89.0%**
VGG Fine-Tune* [34]	89.9%
AURF* [26]	92.1%
CFR-CURL-ELM	**92.9%**
CFR-CURL-ELM†	**93.5%**
AUDN* [75]	93.7%
Zero-bias CNN*† [31]	96.4%
FN2EN*† [34]	96.8%

4.2 Continuous Affect Estimation

4.2.1 Datasets and Evaluation Metrics

To assess our proposed algorithms for the task of continuous affect estimation, we use two datasets of the *Audio-Visual Emotion Challenge* (AVEC) 2012 and 2014 datasets. In the following, we briefly present the datasets and the evaluation metrics we use for assessing our algorithm.

The AVEC 2012 dataset [15] contains video sequences of people interacting with virtual agents. Based on the SEMAINE dataset [76], a benchmarking dataset for naturalistic video and audio of human-agent interactions, the AVEC 2012 dataset is annotated with four affect dimensions (valence, arousal, power and expectation). However, based on the computational framework we propose, we only consider the valence and arousal dimensions. The dataset is subdivided into two sub-challenges. The first sub-challenge is the *fully continuous sub-challenge* (FCSC), involving fully continuous affect estimation, where the level of affect has to be estimated for every moment of recording. The second sub-challenge is the *word-level sub-challenge* (WLSC), involving affect estimation at word-level and only when the recorded user is speaking. In this paper, we use the FCSC and estimate the affective state (valence and arousal) at each frame of the recording. It consists of 63 baseline videos, 31

for training and 32 for development. Each baseline video is recorded at 49.979 frames per second at a spatial resolution of 780×580 pixels.

The aim of the AVEC 2014 dataset is two-fold: (i) estimate the continuous values of the affective dimensions at each moment in time, and (ii) predict the value of a single self-reported severity of depression indicator for each recording in the dataset. The challenge is therefore organized as two sub-challenges, i.e. the *affect recognition sub-challenge* (ASC) and the *depression recognition sub-challenge* (DSC). As the present paper focuses on the problem of continuous affect estimation, we only consider the ASC. The ASC consists of two sub-tasks: *Northwind*, where the participants read aloud an excerpt of a fable spoken in German language and the *Freeform*, where the participants respond to one of a number of questions, again in the German language. For our evaluation, we mix both tasks, resulting in a total of 300 videos equally split in a 100 training, 100 development and 100 test videos.

For rigorously evaluating different algorithms for continuous affect estimation, the AVEC series use Pearson's *correlation coefficient* (CC) as metric. This metric was used for benchmarking algorithms on AVEC 2012 and 2014 datasets. It measures the linear correlation between the two sequences and has a value between $+1$ and -1. A CC value of $+1$ indicates total positive linear correlation, 0 indicates no linear correlation and -1 indicates total negative correlation. Although the affective state vector $\mathbf{a}$ contains valence, arousal and their high-order derivatives, we will only consider the first two dimensions of the state vector (valence and arousal) for comparison, and calculate the CC metric for each of the two dimensions.

4.2.2 Evaluating the Deep Extended Kalman FIlter

In this experiment we quantitatively evaluate our proposed filtering algorithm and compare it to state-of-the-art methods. The main aspects we seek to assess are the contribution of CFR-CURL as model for facial expression representation on the one hand, and the contribution of non-linear neural network models as transition and observation models for the DEKF algorithm. Following the results reported in our previous work [6], we also compare to the MI-SGP-KF pipeline with 2-nd order and 0-th order derivatives for the AVEC 2012 and AVEC 2014 datasets respectively. Tables 3 and 4 summarize the quantitative results on the AVEC 2012 and AVEC 2014 datasets respectively. To be consistent with baseline and state-of-the-art results reported in other works, all the processing pipelines are evaluated using the average CC metric over all the testing videos.

For AVEC 2012, we recall that our baseline MI-SGP-KF method yields very competitive results. It outperforms the state-of-the-art results [77], [78], [79], [80] by more than 10% on average. Similar to our Bayesian filtering strategy, the method presented in [78] uses particle filtering, but surprisingly obtains a much lower score than our solution based on Kalman filtering. To assess the added value of the CFR-CURL processing pipeline for facial expression representation, we combine it with a linear KF and observe a little increase (1%) in CC w.r.t. MI-SGP-KF. Note that, here, we do not use a bag of frames but only compensate for the reaction lag, by re-aligning the frames according to the findings reported in [5]. As input representations $\mathbf{x}_k$, we use whitened canonical face representations followed by the same 25c5-2000f-300f CNN pipeline as presented in figure 6. This CNN model was first trained on the entire CK+ dataset to capture the relevant information related to facial expressions. We then removed the ELM hidden-to-output linear layer and only used the 300-dimensional hidden layer as final representation, hence $\mathbf{y}_k = \mathbf{h}_{\text{CNN}}(\mathbf{x}_k) \in \mathbb{R}^{300}$. This CNN mapping is then used in algorithm 1 for testing our proposed CFR-CURL-DEKF pipeline. For the DEKF, we used transitions and observations models with 20 neurons each. From table 3 we can see that the CFR-CURL-DEKF pipeline performs the best, i.e. an average CC of 54%.

TABLE 3: Quantitative comparison on AVEC 2012: Pearson's *correlation coefficient* (CC) on development data set for the FCSC challenge. Second and third column denote the average CC scores on arousal and valence respectively. Last column denotes the average CC score over arousal and valence.

Method	Arousal	Valence	Mean
Baseline [15]	0.15	0.21	0.18
MI-SGP [6]	0.28	0.31	0.30
CCRF [77]	0.34	0.34	0.34
Video PF [78]	0.31	0.37	0.34
CFER [79]	0.30	0.41	0.36
Multiscale Dynamic Cues [80]	0.51	0.31	0.41
3D Model-Based [81]	0.56	0.45	0.51
MI-SGP-KF [6]	0.53	0.51	0.52
CFR-CURL-KF	**0.55**	**0.50**	**0.53**
CFR-CURL-DEKF	**0.54**	**0.53**	**0.54**

TABLE 4: Quantitative comparison on AVEC 2014: Pearson's *correlation coefficient* (CC) on development data set for the ASC challenge. Second and third column denote the average CC scores on arousal and valence respectively. Last column denotes the average CC score over arousal and valence.

Method	Arousal	Valence	Mean
MI-SGP [6]	0.34	0.27	0.31
Baseline [82]	0.41	0.36	0.39
MI-SGP-KF [6]	0.56	0.51	0.54
CFR-CURL-KF	**0.56**	**0.53**	**0.55**
CFR-CURL-DEKF	**0.61**	**0.55**	**0.58**

For the AVEC 2014 dataset, a similar experiment has been conducted and summarized in table 4. We observe an added value of CFR-CURL for facial expression representation and DEKF for continuous affect estimation. The CFR-CURL representations combined with KF result in an increase in CC (2%) compared to MI-SGP combined with KF. When combined with DEKF (using 20 neurons), we achieve the best average CC of 58%. The conclusion for the AVEC 2012 and AVEC 2014 datasets are therefore consistent: (i) the CFR-CURL pipeline is effective in terms of facial expression representation and (ii) non-linear filtering (DEKF) is effective in terms of continuous affect estimation from facial expressions.

5 Conclusion

Based on the computational framework presented in our previous work [6], we've identified an important gap in terms of computational framework for designing

information-processing systems capable estimating continuous affect from facial expressions, which is the main motivation of this paper. To this end, we've leveraged the synergistic combination between the Bayesian filtering and the deep learning paradigms. We've advanced the state-of-the-art by proposing a novel *canonical face representation* (CFR) algorithm, a novel *convex unsupervised representation learning* (CURL) algorithm, and a novel *deep extended Kalman filtering* (DEKF) algorithm. The performance of the resulting CFR-CURL-DEKF algorithmic framework was empirically evaluated on publicly available benchmark datasets for facial expression recognition (CK+) and continuous affect estimation (AVEC 2012 and AVEC 2014), for which we obtain state-of-the-art results. We can summarize our findings as follows: (i) CFR is necessary for the task of facial expression analysis:, (ii) CURL is sufficient for the task of facial expression representation, and (iii) DEKF is necessary for the task of continuous affect estimation from facial expressions.

ACKNOWLEDGMENTS

This work is supported by the Agency for Innovation by Science and Technology in Flanders (IWT) – PhD grant nr. 131814, the VUB Interdisciplinary Research Program through the EMO-App project, the Chinese Scolarship Council (CSC) through grants nr. 201506290085 and nr. 201706290115, and the National Natural Science Foundation of China (grant 61273265).

REFERENCES

[1] J.-M. Fernández-Dols and C. Crivelli, "Recognition of facial expressions: Past, present, and future challenges," in *Understanding Facial Expressions in Communication*. Springer, 2015, pp. 19–40.

[2] M. C. Oveneke, I. Gonzalez, W. Wang, D. Jiang, and H. Sahli, "Monocular 3d facial information retrieval for automated facial expression analysis," in *IEEE 6th International Conference on Affective Computing and Intelligent Interaction*, 2015, pp. 623–629.

[3] O. Rudovic, M. A. Nicolaou, V. Pavlovic, R. Walecki, O. Rudovic, V. Pavlovic, M. Pantic, S. Eleftheriadis, O. Rudovic, M. Pantic *et al.*, "Machine learning methods for social signal processing," *Social Signal Processing*, vol. 30, pp. 469–484, 2014.

[4] A. Metallinou and S. Narayanan, "Annotation and processing of continuous emotional attributes: Challenges and opportunities," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, 2013, pp. 1–8.

[5] S. Mariooryad and C. Busso, "Correcting time-continuous emotional labels by modeling the reaction lag of evaluators," *IEEE Transactions on Affective Computing*, p. 1, 2014.

[6] M. Oveneke, I. Gonzalez, V. Enescu, D. Jiang, and H. Sahli, "Leveraging the bayesian filtering paradigm for vision-based facial affective state estimation," *IEEE Transactions on Affective Computing*, 2017.

[7] Z. Chen, "Bayesian filtering: From kalman filters to particle filters, and beyond," *Statistics*, vol. 182, no. 1, pp. 1–69, 2003.

[8] J. A. Russell, "Core affect and the psychological construction of emotion." *Psychological review*, vol. 110, no. 1, p. 145, 2003.

[9] Z. Huang and J. Epps, "An investigation of emotion dynamics and kalman filtering for speech-based emotion prediction," *Proc. Interspeech 2017*, pp. 3301–3305, 2017.

[10] D. Marr, *Vision: A Computational Investigation Into the Human Representation and Processing of Visual Information*. MIT Press, 2010.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[12] M. C. Oveneke, M. Aliosha-Perez, Y. Zhao, D. Jiang, and H. Sahli, "Efficient convolutional auto-encoding via random convexification and frequency-domain minimization," in *NIPS 2016 International Workshop on Efficient Methods for Deep Neural Networks (EMDNN)*, 2016.

[13] G. Alain and Y. Bengio, "What regularized auto-encoders learn from the data-generating distribution," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3563–3593, 2014.

[14] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*. IEEE, 2010, pp. 94–101.

[15] B. Schuller, M. Valstar, F. Eyben, R. Cowie, and M. Pantic, "AVEC 2012: the continuous audio/visual emotion challenge," in *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, 2012, pp. 449–456.

[16] M. Valstar, B. Schuller, K. Smith, T. Almaev, F. Eyben, J. Krajewski, R. Cowie, and M. Pantic, "Avec 2014: 3d dimensional affect and depression recognition challenge," in *Proceedings of the 4th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2014, pp. 3–10.

[17] J.-M. Fernández-Dols, F. Sanchez, P. Carrera, and M.-A. Ruiz-Belda, "Are spontaneous expressions and emotions linked? an experimental test of coherence," *Journal of Nonverbal Behavior*, vol. 21, no. 3, pp. 163–177, 1997.

[18] A. J. Fridlund, "What do facial expressions express?" 2001.

[19] M. G. Calvo and L. Nummenmaa, "Perceptual and affective mechanisms in facial expression recognition: An integrative review," *Cognition and Emotion*, pp. 1–26, 2015.

[20] P. Ekman and W. V. Friesen, *Unmasking the face: A guide to recognizing emotions from facial clues*. Ishk, 2003.

[21] E. L. Rosenberg and P. Ekman, "Coherence between expressive and experiential systems in emotion," *Cognition & Emotion*, vol. 8, no. 3, pp. 201–229, 1994.

[22] J. A. Russell, "Emotion, core affect, and psychological construction," *Cognition and Emotion*, vol. 23, no. 7, pp. 1259–1283, 2009.

[23] K. A. Lindquist and M. Gendron, "Whats in a word? language constructs emotion perception," *Emotion Review*, vol. 5, no. 1, pp. 66–71, 2013.

[24] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[25] S. Rifai, Y. Bengio, A. Courville, P. Vincent, and M. Mirza, "Disentangling factors of variation for facial expression recognition," *Computer Vision–ECCV 2012*, pp. 808–822, 2012.

[26] M. Liu, S. Li, S. Shan, and X. Chen, "Au-aware deep networks for facial expression recognition," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*. IEEE, 2013, pp. 1–6.

[27] P. SMOLENSKY, "Information processing in dynamical systems: foundations of harmony theory," *Parallel distributed processing: explorations in the microstructure of cognition*, pp. 194–281, 1986.

[28] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[29] G. E. Hinton, "A practical guide to training restricted boltzmann machines," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 599–619.

[30] P. Liu, S. Han, Z. Meng, and Y. Tong, "Facial expression recognition via a boosted deep belief network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1805–1812.

[31] P. Khorrami, T. Paine, and T. Huang, "Do deep neural networks learn facial action units when doing expression recognition?" in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 19–27.

[32] X. Zhao, X. Liang, L. Liu, T. Li, Y. Han, N. Vasconcelos, and S. Yan, "Peak-piloted deep network for facial expression recognition," in *European Conference on Computer Vision*. Springer, 2016, pp. 425–442.

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[34] H. Ding, S. K. Zhou, and R. Chellappa, "Facenet2expnet: Regularizing a deep face recognition net for expression recognition," in *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*. IEEE, 2017, pp. 118–126.

[35] F. Weninger, F. Ringeval, E. Marchi, and B. W. Schuller, "Discriminatively trained recurrent neural networks for continuous dimensional emotion recognition from audio." in *IJCAI*, 2016, pp. 2196–2202.

[36] H. T. Siegelmann and E. D. Sontag, "Turing computability with neural nets," *Applied Mathematics Letters*, vol. 4, no. 6, pp. 77–80, 1991.

[37] K. Brady, Y. Gwon, P. Khorrami, E. Godoy, W. Campbell, C. Dagli, and T. S. Huang, "Multi-modal audio, video and physiological sensor learning for continuous emotion prediction," in *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2016, pp. 97–104.

[38] P. Tzirakis, G. Trigeorgis, M. A. Nicolaou, B. Schuller, and S. Zafeiriou, "End-to-end multimodal emotion recognition using deep neural networks," *arXiv preprint arXiv:1704.08619*, 2017.

[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[40] B. Sun, S. Cao, L. Li, J. He, and L. Yu, "Exploring multimodal visual features for continuous affect recognition," in *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2016, pp. 83–88.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[42] M. A. Nicolaou, H. Gunes, and M. Pantic, "Continuous prediction of spontaneous affect from multiple cues and modalities in valence-arousal space," *Affective Computing, IEEE Transactions on*, vol. 2, no. 2, pp. 92–105, 2011.

[43] T. Hassner, S. Harel, E. Paz, and R. Enbar, "Effective face frontalization in unconstrained images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4295–4304.

[44] C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic, "Robust statistical frontalization of human and animal faces," *International journal of computer vision*, vol. 122, no. 2, pp. 270–291, 2017.

[45] B. Amberg, R. Knothe, and T. Vetter, "Expression invariant 3d face recognition with a morphable model," in *Automatic Face & Gesture Recognition, 2008. FG'08. 8th IEEE International Conference on*. IEEE, 2008, pp. 1–6.

[46] R. Rosipal and N. Krämer, "Overview and recent advances in partial least squares," in *Subspace, Latent Structure and Feature Selection*. Springer, 2006, pp. 34–51.

[47] S. de Jong, "Simpls: an alternative approach to partial least squares regression," *Chemometrics and intelligent laboratory systems*, vol. 18, no. 3, pp. 251–263, 1993.

[48] A.-L. Boulesteix and K. Strimmer, "Partial least squares: a versatile tool for the analysis of high-dimensional genomic data," *Briefings in bioinformatics*, vol. 8, no. 1, pp. 32–44, 2007.

[49] T. Bie, N. Cristianini, and R. Rosipal, "Eigenproblems in pattern recognition," *Handbook of Geometric Computing*, pp. 129–167, 2005.

[50] I. Matthews and S. Baker, "Active appearance models revisited," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 135–164, 2004.

[51] S. R. Livingstone, K. Peck, and F. A. Russo, "Ravdess: The ryerson audio-visual database of emotional speech and song," in *Annual meeting of the canadian society for brain, behaviour and cognitive science*, 2012, pp. 205–211.

[52] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," vol. 35, no. 8, pp. 1798–1828, 2013.

[53] P. P. Brahma, D. Wu, and Y. She, "Why deep learning works: A manifold disentanglement perspective," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 10, pp. 1997–2008, Oct 2016.

[54] Y. Zhao, M. C. Oveneke, D. Jiang, and H. Sahli, "A video prediction approach for animating single face image," *Multimedia tools and applications (Accepted for Publication)*, 2018.

[55] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 609–616.

[56] Y. Bengio, "Deep learning of representations: Looking forward," in *Statistical Language and Speech Processing*. Springer, 2013, pp. 1–37.

[57] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Ng, "Multimodal deep learning," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 689–696.

[58] J. Tang, C. Deng, and G.-B. Huang, "Extreme learning machine for multilayer perceptron," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 4, pp. 809–821, 2016.

[59] L. C. Liyanaarachchi, Y. Yang, G. Huang, and Z. Zhang, "Dimension reduction with extreme learning machine." *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*, 2016.

[60] A. Rahimi and B. Recht, "Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning," in *Advances in neural information processing systems*, 2009, pp. 1313–1320.

[61] D. W. Kammler, *A first course in Fourier analysis*. Cambridge University Press, 2007.

[62] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.

[63] A. Hefny, D. Needell, and A. Ramdas, "Rows vs. columns: Randomized kaczmarz or gauss-seidel for ridge regression," *arXiv preprint arXiv:1507.05844*, 2015.

[64] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach (3rd edition)*. Prentice Hall, 2009.

[65] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[66] D.-Y. Huang and W. Sun, "A comparison of SVM and asymmetric simpls in emotion recognition from naturalistic dialogues," in *Circuits and Systems (ISCAS), 2012 IEEE International Symposium on*, 2012, pp. 874–877.

[67] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, and D. N. Metaxas, "Learning active facial patches for expression analysis," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2562–2569.

[68] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, "Recognizing facial expression: machine learning and application to spontaneous behavior," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 568–573.

[69] X. Feng, M. Pietikäinen, and A. Hadid, "Facial expression recognition based on local binary patterns," *Pattern Recognition and Image Analysis*, vol. 17, no. 4, pp. 592–598, 2007.

[70] M. Liu, S. Shan, R. Wang, and X. Chen, "Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1749–1756.

[71] M. Liu, S. Li, S. Shan, R. Wang, and X. Chen, "Deeply learning deformable facial action parts model for dynamic expression analysis," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 143–157.

[72] H. Jung, S. Lee, S. Park, I. Lee, C. Ahn, and J. Kim, "Deep temporal appearance-geometry network for facial expression recognition," *arXiv preprint arXiv:1503.01532*, 2015.

[73] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*. IEEE, 2016, pp. 1–10.

[74] K. Sikka, G. Sharma, and M. Bartlett, "Lomo: Latent ordinal model for facial analysis in videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5580–5589.

[75] M. Liu, S. Li, S. Shan, and X. Chen, "Au-inspired deep networks for facial expression feature learning," *Neurocomputing*, vol. 159, pp. 126–136, 2015.

[76] G. McKeown, M. Valstar, R. Cowie, M. Pantic, and M. Schroder, "The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent," *IEEE Transactions on Affective Computing*, vol. 3, no. 1, pp. 5–17, 2012.

[77] T. Baltrusaitis, N. Banda, and P. Robinson, "Dimensional affect recognition using continuous conditional random fields," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, 2013, pp. 1–8.

[78] A. Savran, H. Cao, M. Shah, A. Nenkova, and R. Verma, "Combining video, audio and lexical indicators of affect in spontaneous conversation via particle filtering," in *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, 2012, pp. 485–492.

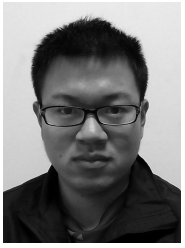
[79] C. Soladié, H. Salam, C. Pelachaud, N. Stoiber, and R. Séguier, "A multimodal fuzzy inference system using a continuous facial expression representation for emotion detection," in *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, 2012, pp. 493–500.

[80] J. Nicolle, V. Rapp, K. Bailly, L. Prevost, and M. Chetouani, "Robust continuous prediction of human emotions using multi-

scale dynamic cues," in *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, 2012, pp. 501–508.
[81] H. Chen, J. Li, F. Zhang, Y. Li, and H. Wang, "3d model-based continuous emotion recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1836–1845.
[82] M. Valstar, B. Schuller, K. Smith, T. Almaev, F. Eyben, J. Krajewski, R. Cowie, and M. Pantic, "Avec 2014: 3d dimensional affect and depression recognition challenge," in *Proceedings of the 4th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2014, pp. 3–10.

Meshia Cédric Oveneke received his BScEng degree in Electronics & Information Technology and MScEng degree in Computer Science (Artificial Intelligence) with great distinction from the Vrije Universiteit Brussel (VUB), Belgium in 2011 and 2013, respectively. He is currently working toward a PhD degree in Engineering Sciences at the Joint VUB-NPU Audio-Visual Signal Processing (AVSP) Laboratory of the VUB Electronics & Informatics (ETRO) department, under the supervision of Professor Hichem Sahli. He is a teaching assistant in computer vision and his current research focuses on developing machine learning based approaches to audio-visual signal processing, applied to affective computing and behavior analysis. His main research interest includes image, video and speech processing, large-scale optimization, machine learning and Bayesian inference.

Yong Zhao received his Bachelor and Master degree in Computer Science and Technology and Computer Application Technologies from Northwestern Polytechnical University (NPU), Xi'an China in 2010 and 2013 respectively. He is currently working toward a PhD degree in Engineering Sciences at the Joint VUB-NPU Audio-Visual Signal Processing (AVSP) Laboratory of the VUB Electronics & Informatics (ETRO) department, under the supervision of Professor Hichem Sahli and Dongmei Jiang. His main research interest includes facial expression synthesis, facial animation and facial expression analysis. He is also interested in image processing, machine learning and human computer interaction.



Ercheng Pei received the Master degree from Northwestern Polytechnical University, China in 2015 in computer science and technology. He is currently a Ph.D student at the School of Computer Science, Northwestern Polytechnical University, and as a visiting student at the Dept. of Electronics and Informatics (ETRO), Vrije Universiteit Brussel (VUB). His research interest focuses on emotion recognition and machine learning.

Dongmei Jiang received her BEng and Master degree in Automatic Control, and PhD degree in Computer Science and Technology from the Northwestern Polytechnical University (NPU), Xi'an China in 1994, 1997 and 2000, respectively. Since then she was affiliated with NPU, where she was appointed as professor of Computer Science and Technology in 2010. She was a visiting scholar at the Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel (VUB), Belgium, from 2001 to 2002, and from 2006 to 2007, respectively. Since 2005, she has been the NPU's team coordinator of the Joint NPU-VUB Audio Visual Signal Processing (AVSP) Lab. Her research has focused on multi-modal affective computing, including emotion recognition from speech, facial expression and body gesture, as well as expressional facial animation synthesis. She is the corresponding author of the winner paper of the Audio Visual$^+$ Emotion Challenge in 2015.

Hichem Sahli is a professor in computer vision and machine learning in the department of Electronics and Informatics (ETRO), Vrije Univeristeit Brussel (VUB), and group-coordinator at the Interuniversitair Micro-Elektronica Centrum vzw (IMEC). He coordinates the Joint VUB-NPU Audio-Visual Signal Processing (AVSP) laboratory. AVSP deals with applied and theoretical problems related to computer vision, machine learning, signal, audio and image processing, for applications linked to affective computing and multi-modal interaction. Sahli's research has focused on computer vision and machine learning, especially in the areas of object detection and tracking, recognition, shape reconstruction, and image segmentation. His work deals with the development of algorithms, analysis, and novel principles for learning.