Int J Comput Vis
DOI 10.1007/s11263-017-1010-1

Confidence-Weighted Local Expression Predictions for Occlusion Handling in Expression Recognition and Action Unit Detection

Arnaud Dapogny[1] · **Kevin Bailly**[1] · **Séverine Dubuisson**[1]

Received: 11 February 2016 / Accepted: 3 April 2017

Abstract Fully-automatic facial expression recognition-break (FER) is a key component of human behavior analysis. Performing FER from still images is a challenging task as it involves handling large interpersonal morphological differences, and as partial occlusions can occasionally happen. Furthermore, labelling expressions is a time-consuming process that is prone to subjectivity, thus the variability may not be fully covered by the training data. In this work, we propose to train random forests upon spatially-constrained random local subspaces of the face. The output local predictions form a categorical expression-driven high-level representation that we call local expression predictions (LEPs). LEPs can be combined to describe categorical facial expressions as well as action units (AUs). Furthermore, LEPs can be weighted by confidence scores provided by an autoencoder network. Such network is trained to locally capture the manifold of the non-occluded training data in a hierarchical way. Extensive experiments show that the proposed LEP representation yields high descriptive power for categorical expressions and AU occurrence prediction, and leads to interesting perspectives towards the design of occlusion-robust and confidence-aware FER systems.

Communicated by Thomas Brox, Cordelia Schmid.

✉ Arnaud Dapogny
arnaud.dapogny@isir.upmc.fr

Kevin Bailly
kevin.bailly@isir.upmc.fr

Séverine Dubuisson
severine.dubuisson@isir.upmc.fr

[1] UPMC Univ Paris 06, CNRS, UMR 7222, Sorbonne Universités, 75005 Paris, France

Keywords Facial expressions · Action unit · Random forest · Occlusions · Autoencoder · Real-time

1 Introduction

Automatic facial expression recognition (FER) from still images is an ongoing research field which is key to many human–computer applications, such as consumer robotics or social monitoring. To address these problems, a lot of emphasis has been put by the psychological community in order to define models that are both accurate and exhaustive enough to describe facial expressions.

Perhaps one of the most long-standing model for describing the expressions is the discrete categorization proposed by Paul Ekman within his cross-cultural studies (Ekman and Wallace 1971), in which he introduced six universally recognized basic expressions (*happiness*, *anger*, *sadness*, *fear*, *disgust* and *surprise*). Along with a *neutral* state, this has been used as an underlying expression model for most attempts at developing a prototypical expression recognition system (Lucey et al. 2010; Yin et al. 2008). However, this model faces limitations for dealing with spontaneous facial expressions (Zhang et al. 2014), as many of our daily affective behaviors may not be translated in terms of prototypical emotions. Nevertheless, the annotation process is rather intuitive, thus there exists a large corpus of labelled data.

Another approach is the continuous affect representation (Green et al. 1989) that consists in projecting expressions onto a restricted number of latent dimensions. A popular example of such model is the valence/activation (relaxed *vs.* aroused)/power (feeling of control)/expectancy (anticipation) model. It is often simplified as a two-dimensional valence-activation representation. However, using such a low-dimensional embedding of facial expressions can cause

Published online: 08 April 2017

the loss of information. Indeed, expressions such as *surprise* are not represented correctly whereas others can not be separated well (*fear vs. anger*). Last but not least, the annotation process is less intuitive than with the categorical representation.

Finally, an alternative facial expression model is the facial action coding system (FACS) (Ekman and Friesen 1977). It consists in describing facial expressions as a combination of 44 facial muscle activations that are refered to as Action Units (AUs). AUs is a face representation that may be less subject to interpretation. It can theoretically be used in accordance with the so-called Emotional FACS (EMFACS) rules in order to describe a broader range of spontaneous expressions. However, the main drawback of the FACS-coding approach is that the annotation tends to be a time-consuming process. Furthermore, FACS coders have to be highly trained, hence limiting the quantity of available data.

In the meantime, as stated in Zeng et al. (2009), FER from still images is challenging as there exists large variability in the morphology or in the expressiveness of different persons. Furthermore, countless configurations of partial occlusion can occasionally happen (e.g. with hand or accessories). As a result, this variability cannot be fully covered by restricted amounts of data. In this paper, we introduce a new categorical expression-driven representation that we call Local Expression Predictions (LEPs). LEPs can be learned efficiently on the available expression datasets, and find applications for occlusion-robust recognition of Ekman's categorical expressions, as well as for confidence-aware AU detection.

2 Related Work

In this section we review recent approaches covering FER, with an emphasis on methods addressing the problem of partial occlusions. We also describe methods for AU detection.

2.1 Occlusion Handling in Categorical FER

Most recent approaches covering FER from still images work in controlled conditions, on a frontal view and lab-recorded environments (Lucey et al. 2010; Yin et al. 2008). Shan et al. (2009) evaluated the recognition accuracy of Local Binary Pattern (LBP) features. Zhao et al. (2014) designed a unified multitask framework for simultaneously performing facial alignment, head pose estimation and FER. Such approaches showed satisfying results in constrained scenarios, but they can face difficulties on more challenging benchmarks (Dhall et al. 2011).

Eleftheriadis et al. (2015) used disciminative shared Gaussian processes to perform pose-invariant FER. Liu et al. (2015) introduced a deep neural network that learns local features relevant for Action Unit prediction, and use it as an intermediate representation for categorical FER. The authors also studied the use of unlabelled data (Liu et al. 2013) to regularize the network training, further enhancing its predictive capacities for FER in the wild. However, none of these approaches explicitly addresses the problem of facial occlusions that are likely to happen in such unconstrained cases.

Kotsia et al. (2008) studied the impact of human perception of facial expressions under partial occlusions, and the predictive capacities of automated systems thereof. Cotter (2010) used sparse decomposition to perform FER on corrupted images. Ghiasi and Fowlkes (2014) use a discriminative approach for facial feature point alignment under partial occlusions. Those approaches rely on explicitly incorporating synthetic occluded data in the training process, and thus struggle to deal with realistic, unpredicted occluding patterns. Zhang et al. (2014) trained classifiers upon random Gabor-based templates. They evaluated their algorithms on synthetically occluded face images, showing that their approach leads to a better recognition rate when the same occluded examples are used for training and testing. Should this not be the case, unpredicted mouth/eye occlusions still lead to a significant loss of performance. Huang et al. (2012) proposed to automatically detect the occluded regions using sparse decomposition residuals. However, the proposed approach may not be flexible enough, as the occlusion detection only outputs binary decisions, and as the face is explicitly divided into only three subparts (eyes, nose and mouth). This limits the capacities of the method to deal with unpredicted forms of occlusion. Finally, another approach (Ranzato et al. 2011) consists in learning generative models of non-occluded faces. When testing on a partially occluded face image, the occluded parts can be generated back and expression recognition can be performed. The pitfall of such an approach is that it does not allow the use of heterogeneous features (e.g. geometric/appearance) which generally lead to better results.

Finally, some approaches consist in assigning confidence weights to specific face regions. For instance, Rifai et al. (2012) use a boosting scheme to select the most relevant facial features for classification. Zhong et al. (2012) as well as Zhao et al. (2015) use the normalized classification scores to select informative regions. In contrast, we learn trees on randomly selected local subspaces during training, then at test time we use the output of a hierarchical autoencoder network to downweight the triangles that are deemed the most uncertain. Thus, if an algorithm such as the one in Zhong et al. (2012) only selects patches around the mouth to disentangle expressions *surprise* from *happiness* and there happen to be an occlusion in that area, the method is bound to fail. By contrast, our WLS-model can use the remaining information (i.e. trees not using the mouth area) to perform classification.

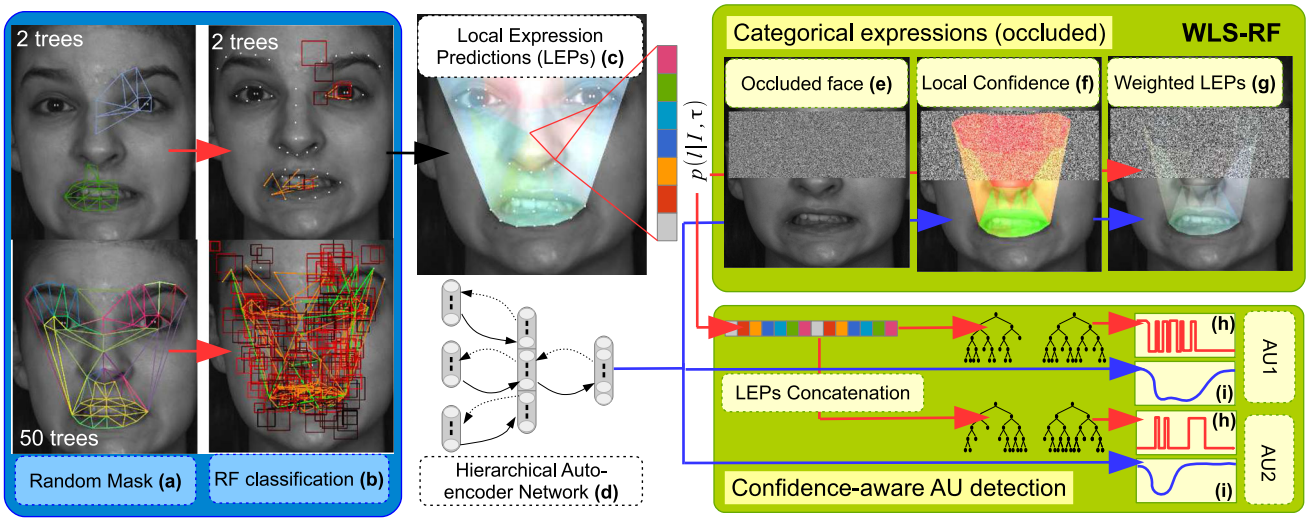

Fig. 1 LEPs and applications to categorical expression recognition, occlusion handling in FER and AU detection. Randomized trees are trained upon local subspaces generated under the form of random facial masks **(a)**, on which binary feature candidates are generated and selected **(b)**. The local predictions outputted by the trees can be aggregated into categorical expression-driven high-level LEP representations **(c)**. Given an occluded face image **(e)**, an occlusion-robust categorical expression prediction can be outputted by weighting LEPs with confidence scores **(f)** given by a hierarchical autoencoder network **(d)**. Furthermore, LEP features can be used to predict AU occurrence **(h)**, for which an AU-specific confidence measurement can be provided **(i)** (Color figure online)

2.2 Action Unit Detection

AU detection is traditionally performed by applying binary classification upon high-dimensional, low-level image descriptors such as LBP or Local Phase Quantification (LPQ) features (Jiang et al. 2011). Sénéchal et al. (2012) proposes to embed heterogeneous (geometric/appearance) features within a multi-kernel SVM framework. These features can be extended to spatio-temporal volumes with the Three Orthogonal Planes (TOP) paradigm, as proposed in Zhang et al. (2014) for LBP-TOP and Jiang et al. (2011) for LPQ-TOP. In the meantime, Chu et al. (2013) introduced a new learning algorithm that personalizes a generic classification framework by attenuating person-specific biases. Nicolle et al. (2015) proposed an AU intensity prediction by a novel multi-task formulation of the metric learning for kernel regression method. However, there seems to be a gap between low-level feature descriptors (e.g. LBP, LPQ, SIFT) and high-end learning algorithms applied to AU detection, that could be filled by learning representations from a large corpus of categorical expression-labelled examples.

Recently, Ruiz et al. (2015) introduced a new framework where categorical expressions are learnt from prior knowledge between this visible task and a set of hidden tasks that correspond to AU activations. Thus, they exploit relationships between those two tasks to learn AU detectors with (SHTL) or without (HTL) FACS-labelled training data, by explicitly combining AUs into categorical expressions *a la* EMFACS. Conversely, we propose to describe AUs as combinations of local expression predictions (LEPs). Furthermore, to the best of our knowledge, this is the first time that a confidence-aware method is proposed for AU detection.

3 Method Overview

In this work, we introduce a new local expression prediction (LEP) representation that can be learned from data labelled with categorical expressions, as described in Fig. 1. During training, local subspaces are generated under the form of random facial masks (a), on which binary candidate features can be selected (b) to train randomized trees. The local LEPs (c) outputted by local subspace random forests can be used for multiple purposes that are depicted below. Furthermore, we also introduce a hierarchical autoencoder network (d), which can be used to capture the local manifold of non-occluded faces around separate aligned feature points. When applied on a potentially occluded face image (e), the reconstruction error outputted by such a network provides a confidence measurement of how close a face region lies from the training data manifold (f), with high and low confidences depicted in green and red respectively. This local confidence measurement can be used to weight LEPs (g) in order to provide an occlusion-robust expression prediction (WLS-RF). Finally, LEPs can be used to predict AU occurrence (h). Once again, the autoencoder network can be used to provide AU-specific confidence measurements (i). Our contributions are thus the following:

1. A Local-subspace RF model that consist in randomized trees learned upon spatially-constrained local subspaces. Those subspaces are generated under the form of random masks covering a specified face region. These local trees are combined to produce high-level expression-driven representations that we refer to as LEPs, which can be used for categorical FER or AU prediction.
2. The use of a hierarchical autoencoder network that learns local non-occluded face manifolds for providing confidence measurements. These confidence scores can be used to enhance robustness to occlusions for categorical FER, as well as to design a confidence-aware AU detection system.

The rest of the paper is organized as follows: in Sect. 4 we describe how we learn the Local Expression Predictions via local subspace random forests (Sect. 4.1) with heterogeneous binary feature candidates (Sect. 4.2). In particular, we explain in Sect. 4.3 how those local representations can be effectively combined and weighted to produce occlusion-robust predictions, and in Sect. 4.4 how LEPs can be used for confidence assessment in AU detection. In Sect. 5 we discuss the proposed autoencoder network architecture (Sect. 5.1), how it is trained to capture the local manifold around facial feature points (Sect. 5.2), and how the reconstruction error of such network can be used for point-wise confidence assessment. Finally, in Sect. 6 we show that our approach significantly improves the state-of-the-art for categorical FER on multiple datasets (described in Sect. 6.1), both on the non-occluded (Sect. 6.3.1) and occluded cases (Sect. 6.3.2). We also demonstrate in Sect. 6.4 the interest of our LEP representation for AU activation prediction and the relevance of the AU-specific confidence measurement. Finally, Sect. 7 provides a conclusion as well as a few perspectives raised in the paper.

4 Local Expression Predictions

Random Forests (RF) is a popular learning framework introduced in the seminal work of Breiman (2001). They have been used to a significant extent in computer vision, and for FER tasks in particular (Zhao et al. 2014; Dapogny et al. 2015), due to their ability to nicely handle high-dimensional data such as images as well as being naturally suited for multiclass classification tasks.

In the classical RF framework, each tree of the forest is grown using a subset of training examples (bagging) and a subspace of the input dimension (random subspace). Individual trees are then grown using a greedy procedure that involves, for each node, the generation of a number of binary split candidates that each consist in a feature ϕ associated with a threshold θ. Each candidate thus defines a partition of the labelled training data. The "best" binary feature is chosen among all features as the one that minimizes an impurity criterion $H_{\phi,\theta}$ (which is generally defined as either the Shannon entropy or the Gini impurity). Then, the above steps are recursively applied for the left and right subtrees with accordingly rooted data until the label distribution at each node becomes homogeneous, where a leaf node is set.

As stated in Breiman (2001), the rationale behind training each tree on a random subspace of the input dimension is that the prediction accuracy of the whole forest depends on both the strength of individual trees and on the independence of the predictions. Thus, by growing individually weaker (e.g. as compared to C4.5) but more decorrelated trees, we can combine these into a more accurate tree collection. Following this idea, we propose an adaptation of the RF framework that uses spatially-defined Local Subspaces (LS) instead of the traditional Random Subspaces (RS). Each tree is trained using a randomly generated, spatially-constrained subspace corresponding to a specific part of the face. Those local models are thus averaged to give rise to Local Expression Predictions (LEPs). Note that this is not the first time that the output classification predictions of RFs are used as features for a subsequent task. For instance, Ren et al. (2014) used local binary features to construct a cascaded feature point alignment method. However, contrary to Ren et al. (2014), we construct our LEP representation by locally averaging predictions and not by directly using the output prediction of the trees. Furthermore, LEPs offer several advantages over using a set of trees defined on the whole face:

1. LEPs can be aggregated for predicting categorical FER. As compared to a global RF, trees of a LS-RF only use features from restricted areas of the face. By combining trees trained on different, randomly generated and spatially-constrained local subspaces, and depending on a locality parameter R, we can find a good compromise between the strength of the individual trees and the increased decorrelation between them. This, in Breiman's term, increase the predictive power of the forests.
2. Given a local confidence measurement (see Sect. 5), we can weight the LEPs for which the pattern lies further from the training data manifold (WLS-RF). For example, in case of occlusion or drastic illumination changes, we can still use the information from the other face subparts to predict the expression.
3. LEPs can be used as an intermediate representation for the task of describing Action Units (AUs). Noteworthy, AU classification could benefit from LEPs trained on larger corpus labelled with categorical expressions, as annotation is less time-consuming than FACS coding.

4.1 Learning Local Trees with Random Facial Masks

The local trees are trained using Algorithm 1. First, we compute the mean shape $\bar{f}$ and the surface $s(\tau(\bar{f}))$ covered by each triangle τ on the mean shape, normalized by the total surface. For each tree t in the forest, we generate a face mask M_t defined over triangles τ. This mask is initialized with a single triangle τ_i randomly selected from the mesh. Then, neighbouring triangles are added until the total surface covered by the selected triangles w.r.t. $\bar{f}$ becomes superior to hyperparameter R, that represents the surface that shall be covered by each tree. Figure 2 provides an illustration of such masks generated on the face mesh. As illustrated on Fig. 2, the choice of R controls the locality of the trees. From a RF perspective, it allows to find a compromise between the strength of individual tree predictors, and the decorrelation between them. Thus, it plays a similar role as the number of features used to set each split node in a traditional RF. We will show experimentally that setting $R = 0.1$ or 0.2 is a good tradeoff in the non-occluded case (Sect. 6.3.1), in addition to bringing substantial improvements in the occluded case (Sect. 6.3.2).

Then, as in the traditional RF induction procedure, we generate a bootstrap by randomly picking 2/3 of the subjects for training tree t. In order to enforce class balance within the bootstraps, we downsample the majority classes. As compared to other methods for balancing RF classifiers (*i.e.* class weighting and upsampling of the minority classes), downsampling leads to similar results while substantially reducing the computational cost, as training is performed on smaller data subsets, as it is explained in Chen et al. (2004). Finally, tree t is grown on a subspace corresponding to the mask M_t.

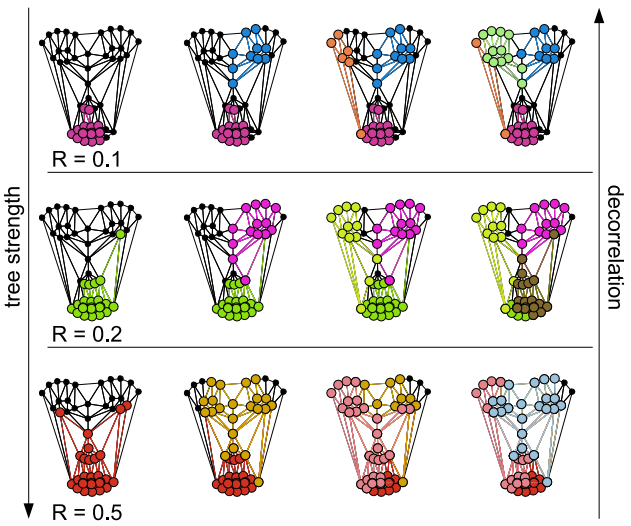

Fig. 2 Masks generated with $R = 0.1$, 0.2 and 0.5 for 1,2,3, 4 trees. Notice how the face is covered by independent masks upon which local trees can be trained. The setting of R allows to find a compromise between tree strength and decorrelation (Color figure online)

Algorithm 1 Training Local Subspace Random Forest

input: images $\mathcal{I}$ with Ids $\mathcal{S}$, labels l and feature points $f(\mathcal{I})$, locality parameter R

```
compute f̄, the mean shape
compute s(τ(f̄)), normalized surface of triangles τ on mean shape
for t = 1 to T do
    randomly select a triangle τ_i
    r ← s(τ_i)
    initialize mask M_t ← {τ_i}
    while r < R do
        draw a list of candidate neighbouring triangles
        randomly select a triangle τ_j from that list
        r ← r + s(τ_j)
        M_t ← M_t ∪ {τ_j}
    end while
    randomly select a fraction S̃_t ⊂ S of subjects
    balance bootstrap S̃_t with downsampling
    grow tree t on bootstrap S̃_t and input subspace M_t
end for
```

output: tree predictors $p_t(l|\mathcal{I})$ with associated masks M_t

4.2 Candidate Feature Generation

Since RFs are non-differentiable predictors, they do not allow end-to-end feature and prediction learning. However, we can generate large numbers of feature candidates on-the-fly during training from generic templates, and select the most relevant to actually split the data. Hence, only the form of the descriptors (e.g. quantifying texture with gradient orientation in the case of HOG) is hand-crafted, and its parameters (e.g. window size and position) can be selected based on the feature relevance. From a RF perspective, we have to extract features from as many independent modalities as possible to bring extra decorrelation to the trees. Furthermore, the feature extraction shall be as fast as possible to keep the runtime low for both training and testing.

To this end, as in Dapogny et al. (2015), we use three feature templates $\phi^{(1)}$, $\phi^{(2)}$ and $\phi^{(3)}$ to generate the split candidates. Binary candidates are then obtained by assigning a threshold θ to each of these instances. More specifically, for each template $\phi^{(i)}$, the upper and lower bounds are estimated from training data beforehand and candidate thresholds are drawn from a uniform distribution in the range of these values. For each node, the "best" binary candidate can be selected as the one that minimizes the impurity criterion $H_{\phi,\theta}$, as in the standard RF induction procedure.

The first two templates are geometric ones, *i.e.* they are generated from the set of N_p facial feature points $f(\mathcal{I})$, provided by an off-the-shelf face alignment algorithm (Xiong and Fernando 2013). The first of these templates is the Euclidean distance between feature points f_a and f_b, normal-

ized w.r.t. intra-ocular distance $iod(f)$ for scale invariance (Eq. 1).

$$\phi^{(1)}_{a,b}(\mathcal{I}) = \frac{||f_a - f_b||_2}{iod(f)} \tag{1}$$

Because the feature point orientation is discarded in feature $\phi^{(1)}$ we use the angles between feature points f_a, f_b and f_c as our second geometric feature $\phi^{(2)}_{a,b,c,\lambda}$. In order to ensure continuity for angles around 0, $\phi^{(2)}$ outputs either the cosine or sine of angle $\widehat{f_a f_b f_c}$, depending on the value of the boolean parameter λ (Eq. (2)):

$$\phi^{(2)}_{a,b,c,\lambda}(\mathcal{I}) = \lambda \cos(\widehat{f_a f_b f_c}) + (1-\lambda)\sin(\widehat{f_a f_b f_c}) \tag{2}$$

As appearance features, we use HOGs for their descriptive power and robustness to global illumination changes. We use integral feature channels as introduced in Dollár et al. (2009). Horizontal and vertical gradients are computed on the image and used to generate 9 feature maps. The first of these contains the gradient magnitude, and the 8 remaining correspond to a 8-bin quantization of the gradient orientation. Then, integral images are computed from these feature maps to output the 9 feature channels. Storing the gradient magnitude within the first channel allows to normalize the histograms as in standard HOG implementations. Thus, we define feature template $\phi^{(3)}_{\tau,ch,sz,\alpha,\beta,\gamma}$ as an integral histogram computed over channel ch within a window of size sz normalized w.r.t. the intra-ocular distance. Such histogram is evaluated at a point defined by its barycentric coordinates α, β and γ w.r.t. vertices of a triangle τ defined over feature points $f(\mathcal{I})$.

4.3 Occlusion-Robust Expression Recognition

When testing, a face image $\mathcal{I}$ is successively rooted left or right for each tree t depending of the outputs of the binary tests stored in the nodes, until it reaches a leaf. The tree t thus outputs a probability vector $p_t(l|\mathcal{I})$ whose components are either 1 for the represented class, or 0 otherwise. Probabilities are then averaged among the T trees (Eq. (3)).

$$p(l|\mathcal{I}) = \frac{1}{T}\sum_{t=1}^{T} p_t(l|\mathcal{I}) \tag{3}$$

Those prediction probabilities are computed similarly for the global RF (RS-RF) and the LS-RF. However, for LS-RF the output probabilities of the trees have some degrees of locality and we can write the above formula as a sum over local probabilities defined for each triangle (Eq. (4)).

$$p(l|\mathcal{I}) = \frac{1}{T}\sum_{\tau} Z_\tau\, p(l|\mathcal{I},\tau) \tag{4}$$

Where $p(l|\mathcal{I},\tau)$ is the Local Expression Prediction (LEP) probability vector associated with triangle τ:

$$p(l|\mathcal{I},\tau) = \frac{1}{Z_\tau}\sum_{t=1}^{T} \frac{\delta(\tau \in M_t)\, p_t(l|\mathcal{I})}{|M_t|} \tag{5}$$

With $\delta(\tau \in M_t)$ being a function that returns 1 if triangle τ belongs to mask M_t, and 0 otherwise. $|M_t|$ is the number of times tree t is used in Eq. (4), and Z_τ is the sum of prediction values for all expression classes l. Thus, a global expression probability is defined by a (normalized) sum of LEPs. Note that those LEP vectors $p(l|\mathcal{I},\tau)$ are not strictly limited to triangle τ but defined within its neighbourhood, with a radius that depends on hyperparameter R.

Moreover, assuming that we have a triangle-wise confidence measurement $\alpha^{(\tau)}$ (see Sect. 5 for how we can provide such a confidence measurement), LEPs can be weighted accordingly to give rise to the Weighted Local Subspace Random Forest model (WLS-RF):

$$p(l|\mathcal{I}) = \frac{\sum_{\tau} \alpha^{(\tau)} Z_\tau\, p(l|\mathcal{I},\tau)}{\sum_{\tau} \alpha^{(\tau)} Z_\tau} \tag{6}$$

This weighting scheme allows to better handle partial occlusions, by downweighting the local RFs associated with the most unreliable appearance patterns, as it will be shown in the experiments.

4.4 Action Unit Detection

4.4.1 From LEPs to Action Units

LEPs are local responses related to categorical facial expressions. Thus, it makes sense to assume that LEPs can somehow be related to AUs and constitute a good high-level representation for AU recognition. To this end, Fig. 3 describes the AU recognition framework, in which LEP vectors corresponding to each triangle are first extracted by training local trees on a categorical expression dataset. The concatenation of all LEP vectors $p(l|\mathcal{I},\tau)$ for every expression l (6 universal expressions plus the neutral one) and triangle τ of the facial mesh gives rise to a $7 \times N_\tau$ feature vector used by a second layer of trees defined for each AU (with N_τ the number of triangles of the facial mesh). Thus, the AU recognition layer is trained on a FACS-labelled dataset using only one feature template $\phi^{(0)}_{l,\tau} = p(l|\mathcal{I},\tau)$, with associated thresholds θ randomly generated from a uniform distribution in the [0; 1] interval.

As illustrated on Fig. 3, we also study the importance of using multiple available expression datasets for learning the first layer of trees (*i.e.* LEP representation). We can

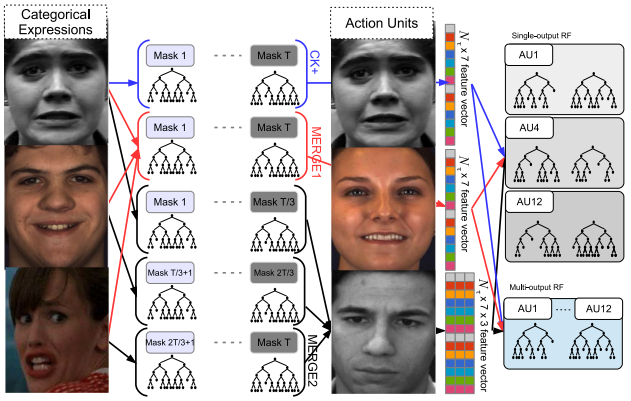

Fig. 3 AU recognition using LEP features. First, LEPs features are learned on data labelled with categorical expressions. LEPs learned on different databases can be aggregated ($M1$ and $M2$ strategies) and used as input to either single-output (SO) or multi-output (MO) RF predictors to perform AU detection

either train the models on a specific categorical expression database, or merge the datasets to learn LEP representation from all the available corpus ($M1$). Finally, we can also learn LEPs separately from the different categorical expression datasets and use a concatenation of the LEP feature vectors as an input for the second (AU prediction) tree layer ($M2$). Section 6.4 shows that those two approaches enhance the predictive power of the AU detection framework. Furthermore, those two strategies can complement each other well. Indeed, $M1$ requires to simultaneously load multiple datasets at training time, $M2$ involves computing multiple LEP features for evaluation. Thus, those two strategies can be combined to fulfil the memory/time requirements.

As it will be shown in the experiments, such a simple, single-output (SO) AU recognition layer already provides decent prediction accuracies. However, as shown in other works on expression recognition (Nicolle et al. 2015), recent approaches such as multi-task formulations (e.g. training a single RF for predicting multiple AUs) can significantly improve performances. In order to efficiently capture the correlations between the different AU prediction tasks, we implemented a recent formulation for learning multi-output (MO) random forests (Henrik 2013) for the AU prediction layer. The main feature of this approach is that it consists in randomly selecting one AU prediction task at each splitting node to compute the Shannon Entropy and select a binary split candidate for that node. As a result, the individual trees are more decorrelated, which in turn increases the predictive power of the whole RF.

4.4.2 Confidence Assessment for AU Prediction

Because AUs are defined locally, chances are that AU activation relatively to an occluded area can not be predicted at all. Thus, we can once again use a confidence measurement to automatically derive a score relative to each AU m. To this end, we define as $N_{l,\tau}^{(m)}$ the number of times that the LEP feature $\phi_{l,\tau}^{(0)}$ was selected for splitting at the root of the trees, among all trees in the forest. This is highlighted on Fig. 4. The reason for exclusively considering features at the root of the trees is that those features are selected from large numbers of training examples, as opposed to features from nodes deeper in the trees, that are essentially more noisy.

Note that, while most approaches focus on describing expressions as a combination of AUs, we can decompose each AU as a set of local expression predictions. For example, for AU1 (inner brow raiser) and AU2 (outer brow raiser), the most relevant LEPs are triangles corresponding to the inner and outer brows, associated with *surprise*, respectively. AU4 (brow lowerer) mainly uses triangles between the eyes associated with expression *anger*. AU9 (nose wrinkler) mainly uses triangles around the nose and cheek, associated with *disgust*. AU12 (lip corner puller) and AU20 (lip stretcher) respectively use triangles corresponding to lip corners with expressions such as *happiness* and *fear*.

We then define the AU-specific confidence measurement α_m for AU m as the sum of confidences $\alpha^{(\tau)}$ of triangles τ of the facial mesh (the measurement of which will be described in Sect. 5), weighted by the proportion of LEP features from that triangle, that are used to describe the activation of AU m:

$$\alpha_m = \frac{\sum_\tau \alpha^{(\tau)} N_{l,\tau}^{(m)}}{\sum_\tau N_{l,\tau}^{(m)}} \tag{7}$$

Thus, the AU-specific confidence measurement is proportional to the confidence of the most useful regions regarding this AU activation prediction. We show in the following section that such simple setting allows to highlight the cases were the AU predictions are deemed unreliable.

5 Manifold Learning of Non-occluded Faces with a Hierarchical Autoencoder Network

In this section, we explain how we can provide a local confidence measurement by modelling the non-occluded local appearance manifold using a hierarchical autoencoder network. Given a number of aligned facial feature points that can be provided by an off-the-shelf alignment algorithm [such as the SDM tracker (Xiong and Fernando 2013)], we use an autoencoder network to model the local face pattern manifold. This network will thus be used to provide a local confidence measurement that is used to weight LEPs for occlusion-robust FER.

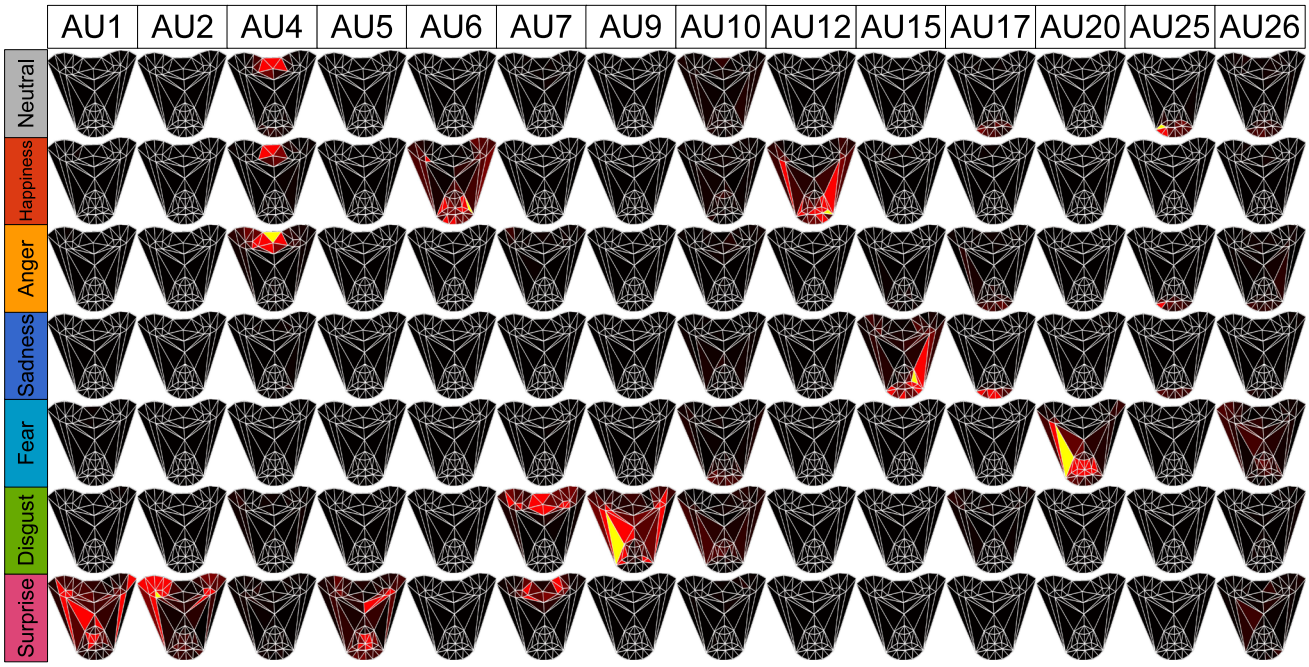

Fig. 4 LEP relevance $N_{l,\tau}^{(m)}$ for AU detection (CK+ database). A high value for an expression l, triangle τ and AU indicates that the corresponding LEP is relevant for discriminating whether AU m is activated or not. For instance, AU 25 can be described by low values of LEPs associated to *neutral* around the mouth, whereas AU2 can be predicted by large values of LEPS associated with *surprise* around the eyes (Color figure online)

5.1 Network Architecture

Autoencoders are a particular type of neural network that can be used for manifold learning. Compared with other approaches such as PCA (Jolliffe 2002), autoencoders offer the advantage to theoretically be able to model complex manifolds using non-linear encoding and regularization criteria, such as denoising (Vincent et al. 2010) or contractive penalties (Rifai et al. 2011). As compared to manifold forests (Pei et al. 2013), autoencoders can be trained on high-dimensional features without falling into the pitfall of low-rank deficiency. Furthermore, they benefit from an efficient training using stochastic gradient descent, as well as the possibility of online fine-tuning for subject-specific calibration.

As shown in Fig. 5, we use a 2-layer architecture, with a first encoding at the feature point level and a second one at the face subpart level. Indeed, the occlusions of neighbouring points are closely related. Thus, by encoding the local texture in a hierarchical way, we can more efficiently capture the relationships between such correlated occlusions. To do so, we first extract HOGs within the neighbourhood of each feature point aligned on the face image $\mathcal{I}$. The choice of modeling a manifold of HOG patterns rather than gray levels stems from the fact that HOGs are used for both the alignment of facial feature points, as well as for LEP generation. Thus, the reconstruction error of these patterns provides a confidence measurement that is relevant for both tasks. Additionally, in order to ensure fast processing, we use integral feature

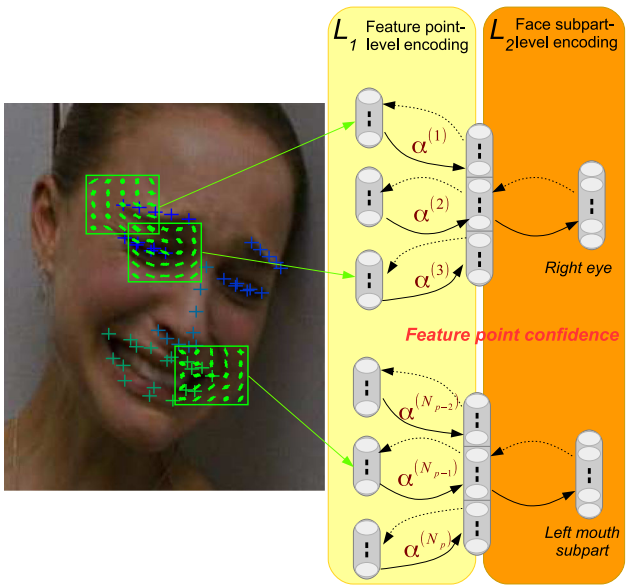

Fig. 5 Architecture of our hierarchical autoencoder network. The network is composed of 2 layers: the first one (L_1) captures the texture variations (HOG descriptors) around the separate aligned feature points. The second one (L_2) is defined over 5 face subparts, each of which embraces multiple points whose appearance variations are closely related. The network outputs a confidence score $\alpha^{(k)}$ for each of the N_p feature points

channels to extract the HOGs. The local descriptor $\Psi^{(k)}$ for a specific feature point k consists in the concatenation of gradient magnitudes and quantized orientation values in 5×5 cells

around this feature point, with a total window size equal to a third of the inter-ocular distance. This descriptor of dimension 225 then feeds the N_p autoencoders (one per feature point) of the first layer (L_1) which are trained to reconstruct non-occluded patterns. Because occlusion of local patterns extracted at the feature point level are not independent (*i.e.* a feature point close to an occluded area is more likely to be occluded itself), we employ a second layer (L_2) of autoencoders, that are trained to reconstruct non-occluded patterns of groups of encoded feature point descriptors. Those groups represent five face subparts (left and right eyes, nose, left and right parts of the mouth) from which the local patterns are closely related. Specifically, L_1 is composed of 125 units for each landmark. L_2 layer for a feature point group contains $65 \times N$ units ($\frac{1}{2}$ compression), where N is 12, 12, 8, 11 and 11 respectively for left/right eye, nose and left/right mouth areas.

5.2 Training the Network

Autoencoders are trained in an unsupervised way, one layer at a time, by optimizing a reconstruction criterion. The input descriptor $\mathbf{\Psi}^{(k)}$ at feature point k is first encoded via the L_1 encoding layer into $h^1(\mathbf{\Psi}^{(k)})$, which is the output of a first neuron layer with a sigmoid activation. This intermediate reconstruction can thus be reconstructed by applying an affine decoder with tied input weights: $\tilde{\mathbf{\Psi}}^{(k)} = g^1 \circ h^1(\mathbf{\Psi}^{(k)})$.

The set of K encoded descriptors $\{h^1(\mathbf{\Psi}^{(k)})\}_{k=1...K}$ associated to feature points $k = 1...K$ that belong to the face subpart m are concatenated to form the input $\boldsymbol{\xi}^{(m)}$ of the layer L_2 for that subpart. Once again, the input of the L_2 layer is successively encoded into an intermediate representation $h^2(\boldsymbol{\xi}^{(m)})$ and decoded in the same way into a reconstructed version $\tilde{\boldsymbol{\xi}}^{(m)} = g^2 \circ h^2(\boldsymbol{\xi}^{(m)})$.

Each layer is trained separately using stochastic gradient descent and backpropagation, by optimizing the squared $\mathcal{L}_2$-loss between an input and its reconstruction through the network. We tried various combinations of training hyperparameters and the best reconstruction results were obtained by applying 15000 stochastic gradient updates with alternating sampling between the expression classes in the databases. Indeed, we want the network to be able to reconstruct local variations of all possible expressive patterns on an equal foot. We also use a constant learning rate of 0.01 as well as a weight decay of 0.001, which provides good results in testing. Finally, we found that adding 25% random masking noise provided satisfying results. From a manifold learning perspective, the goal of using such denoising criterion is to learn to project corrupted examples (e.g. partially occluded ones, which lie further from the manifold) back on the training data manifold. Such example will be reconstructed closer to the training data and its confidence shall be smaller.

5.3 Local Confidence Measurement

Given a face image $\mathcal{I}$, we define the confidence $\alpha^{(k)}(\mathcal{I})$ for point k as a function of the $\mathcal{L}_2$-loss (*i.e.* the reconstruction error) between the HOG pattern $\mathbf{\Psi}(\mathcal{I})$ extracted from this point, and its reconstruction $\tilde{\mathbf{\Psi}}$ outputted by the network, after successively encoding by layers L_1 then L_2, and decoding in the opposite order. By abuse of notation, we have:

$$\alpha^{(k)}(\mathcal{I}) = 1 - \frac{||\mathbf{\Psi}^{(k)} - g^1 \circ g^2 \circ h^2 \circ h^1\left(\mathbf{\Psi}^{(k)}\right)||^2}{\left(||\mathbf{\Psi}^{(k)}|| + ||g^1 \circ g^2 \circ h^2 \circ h^1\left(\mathbf{\Psi}^{(k)}\right)||\right)^2} \quad (8)$$

We used the normalized Euclidean distance as a confidence score as it was directly optimized during training. However, we experimented with other metrics such as RBF, which provided similar results. We introduce a confidence $\alpha^{(\tau)}(\mathcal{I})$ defined over triangles $\tau = \{k_1, k_2, k_3\}$ as:

$$\alpha^{(\tau)}(\mathcal{I}) = \min(\alpha^{(k_1)}(\mathcal{I}), \alpha^{(k_2)}(\mathcal{I}), \alpha^{(k_3)}(\mathcal{I})) \quad (9)$$

As highlighted in the following experiments, this triangle-wise confidence measurement can be used to weight LEPs to enhance the robustness to partial occlusions.

6 Experiments

In this section, we evaluate our approach on several FER benchmarks. Section 6.1 introduces the databases that are used for test. Section 6.2 sums up our experimental protocols and describes hyperparameter settings to ensure reproducibility. In Sect. 6.3.1, we show results on non-occluded data on three publicly available datasets that exhibit various degrees of difficulty. Then, in Sect. 6.3.2, we report results on synthetically occluded images to precisely measure the robustness of our approach to occlusions. In Sect. 6.4 we give results of AU detection, showing that LEPs yield high predictive power for the task of AU detection compared to low-level features or state-of-the-art approaches. We also evaluate the relevance of the AU-specific confidence measurements. Lastly, Sect. 6.5 reports evidence of the real-time capacities of the proposed framework.

6.1 Datasets

6.1.1 Categorical Expressions Datasets

The CK+ or **Extended Cohn-Kanade database** (Lucey et al. 2010) contains 123 subjects, each one displaying some of the 6 universal expressions (*anger*, *happiness*, *sadness*, *fear*, *digust* and *surprise*) plus the non-basic expression *contempt*. Expressions are prototypical and performed in a

controlled lab environment with no head pose variation. As it is done in other approaches, we use the first (*neutral*) and three apex frames for each of the 327 sequences for 8-class FER. As some approaches discard the frames labelled as *contempt*, we also report 7-class accuracy from 309 sequences.

The BU-4D or **BU-4DFE database** (Yin et al. 2008) contains 101 subjects, each one displaying 6 acted categorical facial expressions with moderate head pose variations. Expressions are still prototypical but they are performed with lower intensity and greater variability than in CK+, hence the lower baseline accuracy. Sequence duration is about 100 frames. As the database does not contain frame-wise expression, we manually select neutral and apex frames for each sequence.

The SFEW or **Static Facial Expression in the Wild database** (Dhall et al. 2011) contains 700 images from 95 subjects displaying 7 facial expressions in a real-world environment. Data was gathered from video clips using a semi-automatic labelling process. The strictly person-independent evaluation (SPI) benchmark is composed of two folds of (roughly) same size. As done in other approaches, we report cross-validation results averaged over the two folds.

6.1.2 Action Unit Datasets

The CK+ database is also FACS-annotated, therefore we report results for the recognition of 14 of the most common AUs (AU1,2,4,5,6,7,9,10,12,15,17,20,25,26).

The BP4D database (Zhang et al. 2014) contains 41 subjects. Each subject was asked to perform 8 tasks, each one supposed to give rise to 8 spontaneous expressions (*anger*, *happiness*, *sadness*, *fear*, *digust*, *surprise*, *embarrassment* or *pain*). In Zhang et al. (2014) the authors extracted subsequences of about 20 seconds for manual FACS annotations, arguing that these subsets contain the most expressive behaviors. As done in the litterature (Zhang et al. 2014) we report results for recognition of 12 AUs (1,2,4,6,7,10, 12,14,15,17,23,24). We randomly extract 10,000 images for training and evaluate the AU classifiers on the whole dataset.

The DISFA or **Denver Intensity of Spontaneous Facial Actions** (Mavadati et al. 2013) contains videos of 27 subjects with different ethnicities and genders that were recorded watching a 4-minute emotive video stimulus. Data have been manually labeled frame by frame for 12 AUs (1,2,4,5,6,9,12,15,17,20, 25,26) on a 6-level scale by a human expert, and verified by a second FACS coder. For the purpose of predicting AU occurrence, we consider AU which intensity is below 1 as non-activated. We randomly extract 6292 images for training and test on the 125,832 images.

6.2 Experimental Setup

6.2.1 Evaluation Metrics

For both occluded and non-occluded FER scenarios we use the overall accuracy as a performance metric. For a fair comparison, we report the accuracy for 10-fold subject independent cross-validation, as well as the confusion matrices to show the discrepancies between the expressions.

For AU detection we use the area under the ROC curve (AUC) as a performance metric, as it is widely used in the literature because it is independent of the setting of a decision threshold. Moreover, in order to compare with other sate-of-the-art methods, we also report the F1 and F1-norm scores. The F1-norm score is obtained by normalizing the recall for one AU by a skew factor that is equal to the ratio between the number of positive and negative examples for that AU. For all the experiments, RF classifiers are evaluated with Out-Of-Bag (OOB) error estimate, with bootstraps generated at the subject level to ensure that, for each tree, subjects used for training are not used for testing this specific tree. The OOB error, according to Breiman (2001), is an unbiased estimate of the true generalization error. Moreover, as stated in Bylander (2002) this estimate is generally more pessimistic than traditional (e.g. k-fold or leave-one-subject-out) cross-validation estimates, further reflecting the quality of the results. For AU recognition, LEPs are generated for Out-Of-Bag examples for each tree and AUs are evaluated with OOB error.

6.2.2 Hyperparameter Setting

In order to decrease the variance of the error we train large collections of trees ($T_1 = 1000$ for LEP generation, $T_2 = 50$ for AU detection). For training the local models, we set the locality parameter R to 0.1,which means that each local model uses $1/10$ of the face total surface. Finally, we use 40 $\phi^{(1)}$, 40 $\phi^{(2)}$ and 160 $\phi^{(3)}$ features for learning LEPs, as well as 25 threshold evaluations per features. For AU detection, we examine 100 $\phi^{(0)}$ features at each node, each associated with 25 threshold values. Note however that the values of these hyperparameters (except for R) had very little influence on the performances. This is due to the complexity of the RF framework, in which individually weak trees (e.g. that are grown by only examining a few features per node) are generally less correlated, still outputting decent predictions when combined altogether.

For the occluded scenarios on CK+ and BU4D, the autoencoder networks are trained in a cross-database fashion (*i.e.* training on CK+ and testing on BU4D and vice versa). On SFEW database, we use the autoencoders trained on CK+, as SFEW embraces multiple examples of occluded faces.

6.3 Categorical FER

6.3.1 FER on Non-occluded Images

In Tables 1, 2 and 3 we report the average accuracy obtained by our local subspace random forest (LS-RF) and the confidence-weighted version (WLS-RF). We also compare with standard RF (RS-RF). Generally speaking, classification results of LS-RF are a little better than those of the RS-RF. Indeed, forcing the trees to be local allows to capture more diverse information. RS-RF relies quite heavily on the mouth region, but other areas (e.g. around the eyes, eyebrows and nose regions) may also convey information that can be captured by local models. Figure 6 displays the proportion of top-level features over all triangles of the face area.

Table 1 CK+ database

CK+	7em	8em
LBP (Shan et al. 2009)	88.9[†]	–
CSPL (Zhong et al. 2012)	89.9[†]	–
iMORF (Zhao et al. 2014)	–	90.0
AUDN (Liu et al. 2015)	93.7	92.0
RS-RF	92.6	91.5
LS-RF	93.9	93.1
WLS-RF	**94.3**	**93.4**

Bold values indicate the best results
[†] CK database

Table 2 BU4D database

BU4D	% Acc
BoMW (Xu and Mordohai 20100	63.8
Geometric (Sun and Yin 2008)	68.3
LBP-TOP (Hayat et al. 2012)	71.6
2D FFDs (Sandbach et al. 2011)	73.4
RS-RF	73.0
LS-RF	74.3
WLS-RF	**75.0**

Bold values indicate the best results

Table 3 SFEW database

SFEW	% Acc
PHOG-LPQ (Dhall et al. 2011)	19.0
DS-GPLVM (Eleftheriadis et al. 2015)	24.7
AUDN (Liu et al. 2015)	30.1
Semi-Supervised (Liu et al. 2013)	34.9
RS-RF	35.7
LS-RF	35.6
WLS-RF	**37.1**

Bold values indicate the best results

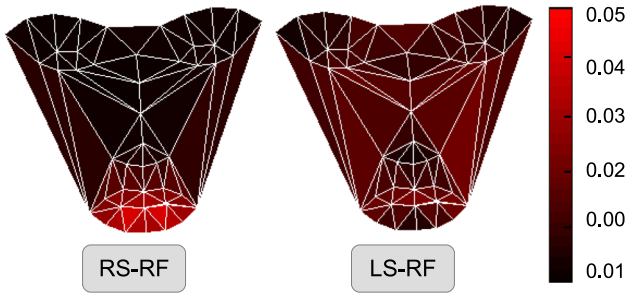

Fig. 6 Proportion of *top-level* (tree root) selected features per *triangle* for categorical FER (CK+ database) (Color figure online)

Table 4 Confusion matrix (CK+-8em)

	ne	ha	an	sa	fe	di	co	su
ne	92.4	0.3	0.9	0.6	1.2	0.6	3.97	0
ha	0	100	0	0	0	0	0	0
an	4.4	0	91.1	0	0	2.3	2.3	0
sa	22.6	0	0	77.4	0	0	0	0
fe	1.3	4	0	0	90.7	0	0	4
di	3.4	0	0.6	0	0	96.1	0	0
co	11.1	0	0	3.7	0	0	85.2	0
su	1.6	0	0	0	0.4	0	1.2	96.8

While more than 90% of the features extracted by RS-RF are concentrated around the mouth, the repartition for LS-RF is more homogeneous. Hence, LS-RF is less prone to a misalignment of the mouth feature points, or to occlusions of the mouth region. Furthermore, weighting the local predictions (WLS-RF) using the confidence score from the autoencoder network allows to enhance the results on the three datasets. The reason is that some subjects may exhibit uncommon morphological traits, occlusion or lighting patterns. As such, more emphasis is put on reliable local patterns, resulting in a better overall accuracy. Moreover, LS-RF and WLS-RF models provide better results compared to state-of-the-art approaches, even though some of these use complex FFD or spatio-temporal features (LBP-TOP), or use additional unlabelled data for regularization (Liu et al. 2013). Note however that the evaluation protocols are different for some of these approaches. For example, authors in Eleftheriadis et al. (2015) only use the texture information.

Tables 4, 5 and 6 show the confusion matrices of WLS-RF on CK+, BU4D and SFEW respectively. Expressions *neutral*, *happy* and *surprise* are mostly correctly recognized, as they involve the most recognizable patterns (smile or eyebrow raise). *Anger* and *disgust* are also accurately recognized on CK+ and BU4D but not so much on SFEW. *Sadness* and *fear* seems to be the most subtle ones, particularly on BU4D and SFEW where those expressions can be misclassified as *surprise* or *happy*, respectively.

Table 5 Confusion matrix (BU4D)

	ne	ha	an	sa	fe	di	su
ne	89.5	0	1.8	4.4	0.9	0.9	2.6
ha	2	89.9	0	0	5	2	1
an	10.1	0	70.7	7.1	2	9.1	1
sa	11	0	15	71	3	0	0
fe	9.8	17.6	2.9	5.9	38.3	11.8	13.7
di	3	4	6.9	1	7.9	73.3	4
su	0	1	0	1	6.2	0	91.8

Table 6 Confusion matrix (SFEW)

	ne	ha	an	sa	fe	di	su
ne	50.2	8.8	9.0	10.0	2.0	16.9	3.1
ha	10.6	67.5	6.2	6.9	2.6	3.5	2.6
an	25.4	16.1	31.3	10.1	3.7	0.9	12.5
sa	21.2	21.2	8.1	22.2	7.1	9.1	11.1
fe	14.2	16.2	13.0	5.0	23.1	7.1	21.3
di	31.3	23.7	10.4	7.1	3.7	15.6	8.2
su	15.4	11.0	12.1	3.3	7.7	6.6	44.0

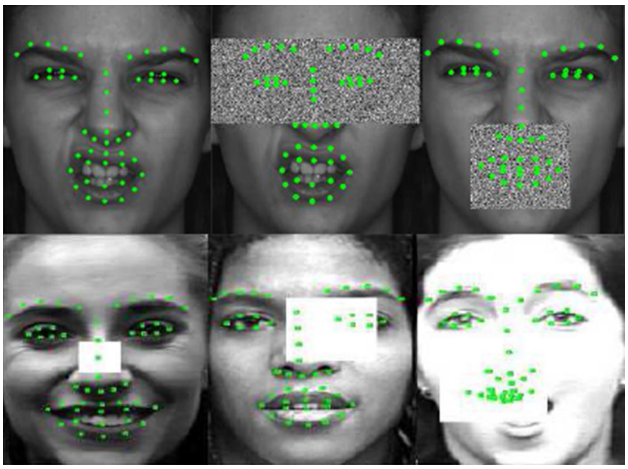

Fig. 7 Examples of occluded faces. *Top row*: non-occluded, eyes occluded, mouth occluded. *Bottom row*: random occlusions (R8, R16, R24) also notice how the presence of an occlusion may have a critical effect on the quality of the feature point alignment

6.3.2 FER on Occluded Face Images

In order to assess the robustness of our system to occlusions, we measured the average accuracy outputted by our models on CK+ (8 expressions) and BU4D (7 expressions) with synthetic occlusions. For each image we used the feature points tracked on non-occluded images to highlight the eyes and mouth regions. We then overlay an occluding pattern (see Fig. 7) either based on the eye or mouth location, or randomly with variable sizes (R8, R16 and R24). We added 20 pixels margins to make sure we cover the whole eyes with eyebrows and mouth region. Finally, we once again align the feature points on the occluded sequences.

Influence of R Graphs of Fig. 8 show the variation of average accuracy *vs.* hyperparameter R that controls the locality of the trees, respectively under eyes and mouth occlusion on CK+ database. Performances of RS-RF fall heavily when the mouth is occluded (from 91.5 to 25.4%), as observed in Zhang et al. (2014). This further proves that the global model relies essentially on mouth features to decipher facial expressions. Forcing the trees to be more local (e.g. setting R to 0.1 or 0.2) allows to capture more diverse cues from multiple facial areas, ensuring more robustness to mouth occlusion. It also explains why LS-RF models with $R = 0.8 - 0.5$ can already be quite robust to eyes occlusions, as the majority of the information used on such models likely comes from mouth area. Nevertheless, on those two occlusion scenarios, WLS-RF achieves a substantially better accuracy than the unweighted local models. Figure 8 also shows the accuracy comparison for both eyes and mouth occlusion scenarios on CK+ and BU4D, with $R = 0.1$. On the two databases, LS-RF is more robust to partial occlusions than RS-RF. Furthermore, WLS-RF also provides better accuracy than both LS-RF and RS-RF. Table 7 draws a comparison between our method and Zhang et al. (2014) with the same protocol. Our approach provides significantly better results than the one in Zhang et al. (2014) in case of severe occlusions (mouth or R24). For instance, the recognition percentage for WLS-RF under mouth occlusion is 72.7 against 30.3% for Zhang et al. (2014) in the case where classifiers are trained on non-occluded faces and tested on occluded ones. This demonstrates the flexibility of the proposed work.

Realistic occlusions our occlusion model is however quite "boring", in the sense that the occluding patterns are not realistic. For that matter, and because there is currently no FER database that includes annotated partial occlusion ground truth, we also present on Fig. 9 qualitative results on more realistic occlusions. Notice how the autoencoder network (learnt on CK+) assign high confidences (green) to non-occluded feature points, whereas examples that lie further from the captured manifold (e.g. because of lighting conditions, self-occlusion with a hand or with an accessory) are given lower values (red). Also note that different regions can vote for different expressions (such as *happy+angry/disgust* or *surprise+happy*). Thus, it would be interesting to study the capacity of a LS-RF trained on categorical FER databases to predict compound expressions (Du et al. 2014).

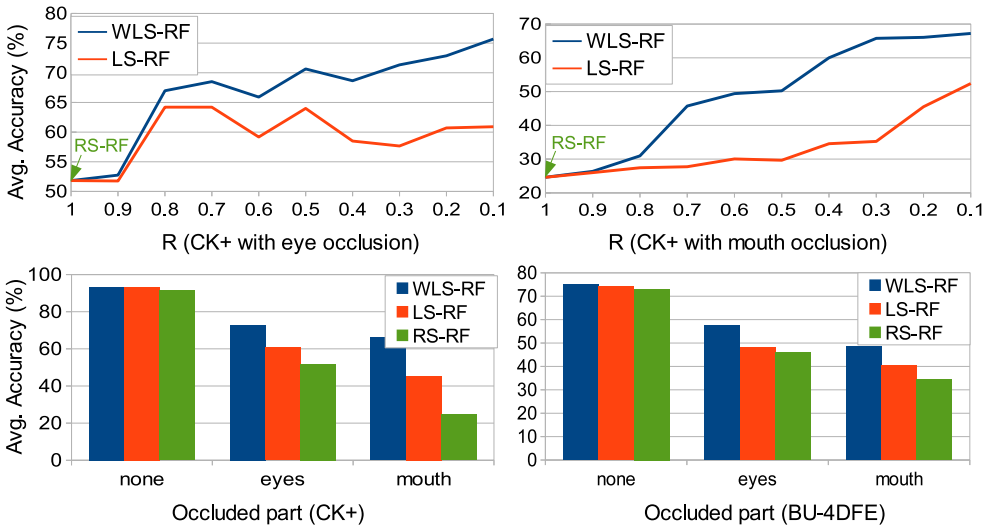

Fig. 8 Accuracy outputted on occluded CK+ and BU4D databases

Table 7 Random and targeted synthetic occlusions on CK+ database

Occlusion	WLS-RF	RGBT (Zhang et al. 2014)
None	**94.3**	**94.4**
R8	**92.2**	**92**
R16	**86.4**	82
R24	**74.8**	62.5
Eyes occluded	**87.9**	**88**
Mouth occluded	**72.7**	30.3

Bold values indicate the best results

6.4 AU Detection

6.4.1 Merging Multiple Datasets

In this section we present results for AU detection using LEP features and a single-output AU prediction layer. Table 8 shows comparison of AUC for the prediction of AU activations on CK+ database obtained with LEPs trained on CK+, BU4D and SFEW databases, as well as models obtained *via* the $M1$ and $M2$ strategies (see Sect. 4.4.2).

For nearly every AU on CK+, the best AUC score is provided by the $M2$ strategy. However LEPs trained on CK+ only as well as the $M1$ strategy also provide good prediction results. LEPs trained solely on BU4D and SFEW seems a bit lackluster, but using the additional categorical expression data in addition to data from CK+ can be beneficial for prediction accuracy. Interestingly, on BP4D, LEPs trained on CK+ only seem to have a slight edge over the two LEPs models trained using all the available data. However, the $M2$ strategy and, to a lesser extent, $M1$ and training on BU4D only, provide close performances. Furthermore, on the DISFA dataset, the $M1$ and the $M2$ LEPs models provide the highest AUC. Overall, the $M2$ and $M1$ models seem to perform better, followed by the models trained on CK+. This proves that AU detection can benefit from additional training data labelled with categorical expressions. Finally, LEPs trained on SFEW did not perform very well, probably due to the fact that the database embraces too much variability for too few training data. Thus, the categorical expressions can not be captured adequately, as can be seen from the low accuracies showed in Sect. 6.3.1.

6.4.2 Comparison with State-of-the-Art Approaches

Table 9 reports the overall best AUC, F1 and F1-norm obtained on the three datasets, using the M2 merging strategy for LEPs and SO/MO AU prediction. It also draws a comparison between the scores obtained using our method and results reported in recent publications involving similar protocols (same databases and sets of AUs, same intensity threshold for AU occurrence on DISFA). First, on the three databases, the MO model is significantly better than SO both in terms of AUC, F1 and F1-norm. Our approach provides significantly better results than JMPL (Zhao et al. 2015) and MCLVM (Eleftheriadis et al. 2015) on CK+. On BP4D, JMPL provides a better F1-norm score. However, it uses a restricted set of highly correlated AUs for capturing relationships within its multi-label learning process. Still, our MO approach provides better results than two recent deep multi-label learning methods, namely MLCNN (Ghosh et al. 2015) and DRML (Zhao et al. 2016) in terms of AUC and F1-score, respectively. Furthermore, on DISFA, our approach provides substantially better AUC than Ghosh et al. (2015). This demonstrates that LEPs learned on large amounts of categorical expression data yield high discriminative power for AU detection tasks. Note that Liu et al. (2015) pointed out the fact that additional AU-labelled data

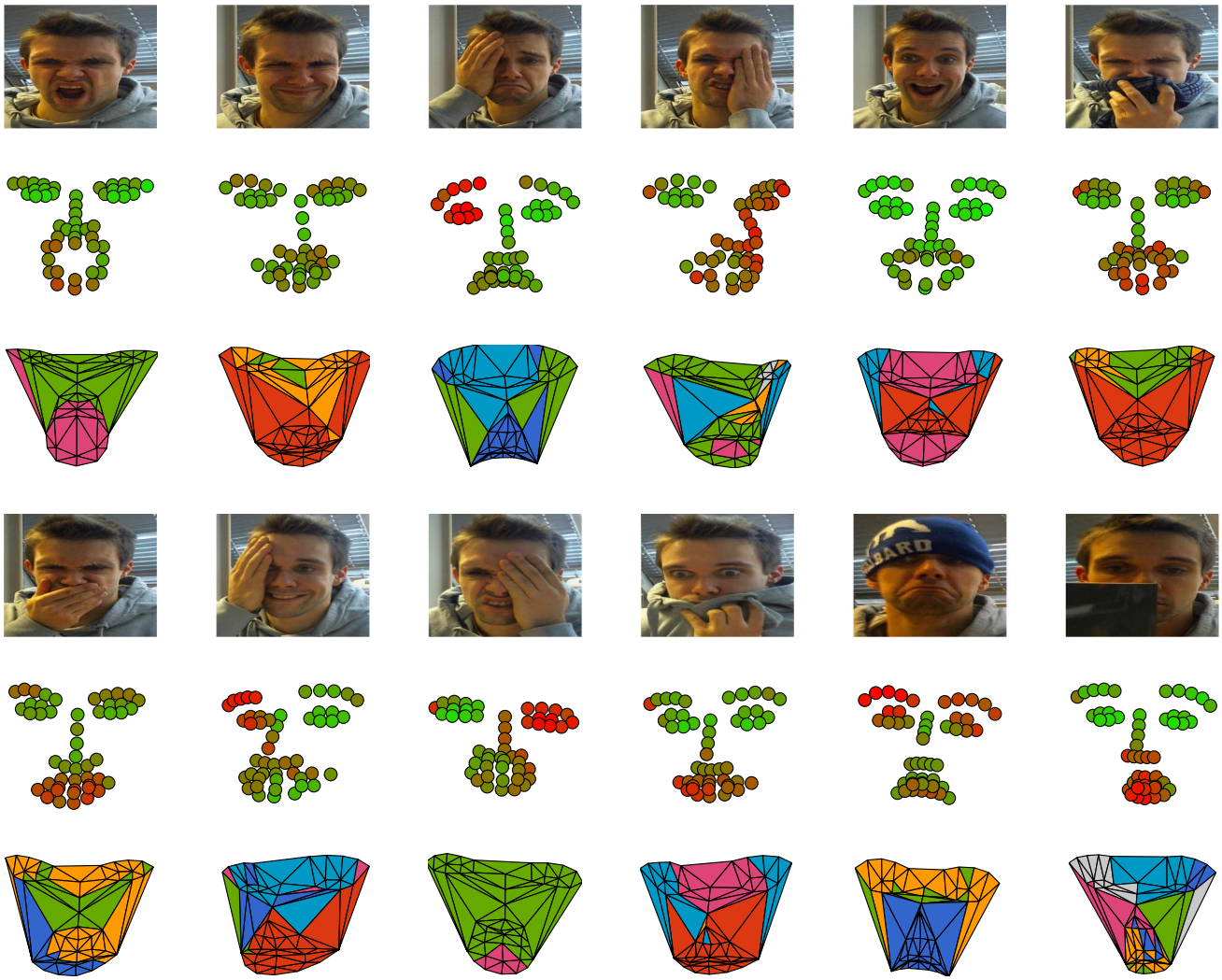

Fig. 9 Examples of local FER under realistic occlusions. *Middle rows*: point-wise confidence scores (*red*: low confidence, *green*: high confidence) *Bottom rows*: maximum LEP values per *triangle* (*gray* for neutral, *red* for *happy*, *yellow* for *angry*, *blue* for *sad*, cyan for *fear*, *green* for *disgust* and magenta for *surprise*) (Color figure online)

can help the training of an expression recognition system. We essentially show that leveraging categorical expression-labelled data using local predictors can also be beneficial for AU prediction. Furthermore, multi-output strategies allow to significantly improve the performance of AU detection systems by efficiently capturing the relationships between those highly correlated tasks.

6.4.3 Relevance of AU Confidence Assessment

In order to assess the relevance of the AU-specific confidence measurement, we evaluated its average value on the occluded versions of the CK+ and BU4D databases generated in Sect. 6.3.2 for occlusion handling in categorical FER. From a general perspective, as can be seen on Fig. 10, low confidence measurements can be observed for AUs from the upper face region on the two scenarios involving eye occlusion. The same holds for AUs from the lower face region and the "mouth occluded" scenario, whereas the confidence scores are significantly higher in the non-occluded case. Interestingly, confidence scores for AU6 (cheek raiser) and, to a lesser extent, AU9 (nose wrinkle), are quite low even in the "mouth occluded" case. Indeed, as can be witnessed on Fig. 3, the confidence measurement for these AUs also use LEP features from the nose and mouth area.

6.5 Computational Load

The proposed framework for occlusion-robust FER (WLS-RF) and AU detection operates in real-time on video streams, even with large tree collections. Table 10 displays the elapsed

Table 8 AUC scores on CK+, BP4D and DISFA databases

	CK+					BP4D					DISFA				
AU	M1	M2	CK+	BU4D	SFEW	M1	M2	CK+	BU4D	SFEW	M1	M2	CK+	BU4D	SFEW
AU1	97.9	**98.4**	98.4	94.7	93.3	59.6	62.7	**63.6**	60.9	52.0	66.1	68.4	**71.3**	57.6	66.6
AU2	98	**98.2**	97.7	97.5	97.2	65.4	64.8	62.3	**66.0**	53.0	53.8	55.2	**67.3**	59.3	59.4
AU4	93.3	**95.4**	94.8	83.1	85.6	**68.7**	63.8	64.4	64.4	55.3	66.7	66.7	**67.3**	64.0	67.6
AU5	94	**97.5**	95.5	93.2	95	–	–	–	–	–	84.2	85.6	73.3	**88.6**	73.7
AU6	95.4	**95.7**	95.5	94.3	94.9	**83.1**	81.8	82.6	78.5	77.1	89.1	86.0	**89.2**	86.8	85.1
AU7	89.1	**90.2**	89.6	88.1	83	**76.8**	75.0	73.6	72.6	65.0	–	–	–	–	–
AU9	97.9	**99.3**	98.7	98.5	94.8	–	–	–	–	–	**79.0**	77.0	75.4	74.0	53.4
AU10	83.7	85.6	**86.5**	78.4	81.7	83.7	**83.8**	83.3	81.0	78.6	–	–	–	–	–
AU12	**97.6**	96	96.2	96	96.5	89.9	**90.0**	89.8	88.0	87.2	**95.5**	92.9	93.6	92.8	91.8
AU14	–	–	–	–	–	65.2	66.4	63.7	**66.5**	64.9	–	–	–	–	–
AU15	**91**	88.9	88.3	79	79.5	56.8	58.4	**58.5**	57.7	56.0	**69.5**	64.5	63.6	68.8	61.7
AU17	93.9	**95.1**	93.4	81.5	86.4	55.8	65.7	**68.9**	65.1	60.6	**67.8**	61.2	53.5	59.1	58.8
AU20	91.9	93.8	**94.5**	88.5	85.8	–	–	–	–	–	**65.0**	58.5	50.2	55.5	61.9
AU23	–	–	–	–	–	50.1	57.2	**60.2**	57.5	54.2	–	–	–	–	–
AU24	–	–	–	–	–	69.6	77.4	**78.2**	77.7	68.4	–	–	–	–	–
AU25	99	**99.1**	98.8	87.1	97.4	–	–	–	–	–	94.8	95.0	94.0	**95.6**	80.0
AU26	75.7	**81.2**	79.7	74.9	73.4	–	–	–	–	–	79.3	**81.4**	75.6	78.5	71.5
Avg	92.7	**93.7**	93.4	88.2	88.9	68.8	70.6	**70.8**	69.6	64.3	**75.9**	74.4	72.9	73.4	69.3

Bold values indicate the best results

Table 9 Comparison with other methods

	CK + (9AUs)			BP4D(12AUS)			DISFA(8AUS)	
Method	AUC	F1	F1-norm	AUC	F1	F1-norm	AUC	F1
MLCNN (Ghosh et al. 2015)	–	–	–	72.3	–	–	78.9	–
JMPL (Zhao et al. 2015)	–	–	78	–	–	**68**	–	–
MCLVM (Eleftheriadis et al. 2015)	–	74.31	–	–	–	–	–	–
DRML (Zhao et al. 2016)	–	–	–	–	48.3	–	–	–
This work, LEP-SO	93.8	72.2	84.8	70.2	52.5	60.3	77.8	44.1
This work, LEP-MO	**95.3**	**78.8**	**86.5**	**72.7**	**55.7**	63.6	**82.4**	**49.1**

Bold values indicate the best results

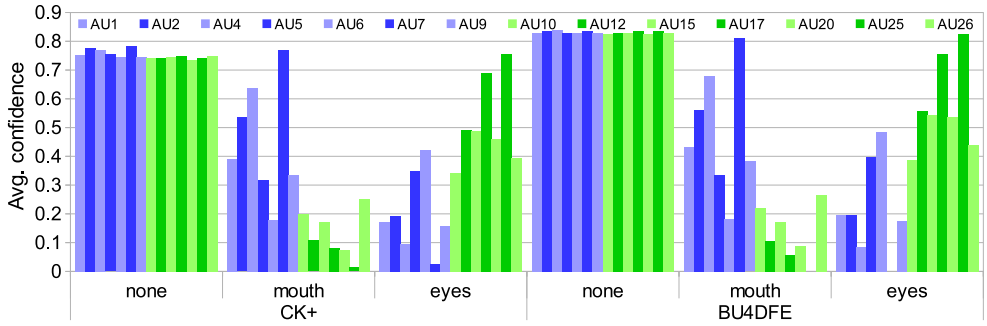

Fig. 10 AU confidence scores outputted on occluded CK+ and BU4D database

time for each step of the evaluation pipeline. The test was performed on an Intel Core I7-4770 CPU on a single-thread C++/OpenCV implementation.

It appears that the feature point alignment and confidence weight generation steps are the bottleneck of the system runtime-wise. However the computational load for the for-

Table 10 Measured evaluation time per processing step (in milliseconds)

Processing step	Time (ms)
Feature point alignment	10
Integral channels computation	2
Confidence weights computation	11
LEP computation (1000 trees)	7
12 AU detection (50 trees)	1
Total	31

mer can be reduced by the use of more efficient alignment algorithms such as the one in Ren et al. (2014). As for the confidence weights, the computation time can be significantly reduced by using a proper multithreading. As it is, the framework already runs at more than 30 fps even with large collections of trees. As for training, learning LEPs with 1000 trees on a big database (BU4D containing more than 8000 face images) took approximately three hours without parallelization. Training the hierarchical autoencoder network took half a day and learning the 12 AU detectors on DISFA database with 50 trees required one hour on the same I7-4770 CPU using a loose C++ implementation. Thus, our approach scales well both in terms of training and testing times, especially when compared to recent deep learning algorithms (Ghosh et al. 2015) for feature representation and learning.

7 Conclusion and Perspectives

In this paper, we introduced a new high-level expression-driven LEP representation. LEPs are obtained from training Random Forests upon spatially defined local subspaces of the face. Extensive experiments on multiple datasets highlight the fact that the proposed representation improves the state-of-the-art for categorical FER and yields useful descriptive power for AU occurrence prediction. Furthermore, we introduced a hierarchical autoencoder network to model the manifold around specific facial feature points. We showed that the provided reconstruction error could effectively be used as a confidence measurement to weight the prediction outputted by the local trees. The proposed WLS-RF framework significantly adds robustness to partial face occlusions.

The ideas introduced in this work open a lot of interesting directions for future works on face analysis. First, note that the confidence weights are representative of the spatially defined local manifold of the training data. Thus, these confidence values can be used to determine which parts of the face are the most reliable in a general way (e.g. to address unusual illumination patterns or head pose variations), and are not limited to occlusion handling. Furthermore, we could inject confidence weights into the feature point alignment framework (Xiong and Fernando 2013) to enhance the robustness of the feature point alignment w.r.t. occlusions. Compared to a discriminative approach using synthetic data (Ghiasi and Fowlkes 2014), our manifold learning approach could in theory more efficiently deal with realistic occlusions. Moreover, the applications of LEPs for AU detection and intensity estimation are multiple. First, it would be interesting to learn LEPs using more expression data such as the datasets introduced in Savran et al. (2008), Wallhoff (2006), possibly with a more complex integration strategy. Also, it could be interesting to investigate the impact of using a more fine-grained facial mesh for FER and AU detection or intensity estimation using LEP representation, as it was done in Jeni et al. (2015) for dense facial feature point alignment. Last but not least, the idea of learning Random Forests upon spatially defined local subspaces instead of random subspaces could be applied (with different settings of the locality parameter) to other face analysis tasks such as face detection and feature point alignment, age or gender prediction. Last but not least, an interesting application would be to study the applicability of a LS-RF trained on categorical expressions to predict the compound facial expressions of emotion (Du et al. 2014).

Acknowledgements This work has been supported by the French National Agency (ANR) in the frame of its Technological Research CONTINT program (JEMImE, project number ANR-13-CORD-0004).

References

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Bylander, T. (2002). Estimating generalization error on two-class datasets using out-of-bag estimates. *Machine Learning*, *48*(1–3), 287–297.

Chen, C., Liaw, A., & Breiman, L. (2004). *Using random forest to learn imbalanced data* (Vol. 110). Technical report. Berkeley: University of California.

Chu, W.-S., De la Torre, F., & Cohn, J. F. (2013). Selective transfer machine for personalized facial action unit detection. In *CVPR* (pp. 3515–3522).

Cotter, S. F. (2010). Sparse representation for accurate classification of corrupted and occluded facial expressions. In *ICASSP* (pp. 838–841).

Dapogny, A., Bailly, K., & Dubuisson, S. (2015) Pairwise conditional random forests for facial expression recognition. In *ICCV*.

Dhall, A., Goecke, R., Lucey, S., & Gedeon, T. (2011). Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In *ICCV Workshops* (pp. 2106–2112).

Dollár, P., Tu, Z., Perona, P., & Belongie, S. (2009). Integral channel features. In *BMVC*.

Du, S., Tao, Y., & Martinez, A. M. (2014). Compound facial expressions of emotion. In *Proceedings of the National Academy of Sciences* (pp. 111).

Ekman, P., & Friesen, W. V. (1977). *Facial action coding system*. Palo Alto: Consulting Psychologists Press.

Ekman, Paul, & Friesen, W. V. (1971). Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, *17*(2), 124.

Eleftheriadis, S., Rudovic, O., & Pantic, M. (2015). Multi-conditional latent variable model for joint facial action unit detection. In *ICCV*.

Eleftheriadis, S., Rudovic, O., & Pantic, M. (2015). Discriminative shared gaussian processes for multiview and view-invariant facial expression recognition. *IEEE Transactions on Image Processing*, *24*(1), 189–204.

Ghiasi, G., & Fowlkes, C. C. (2014). Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model. In *CVPR* (pp. 1899–1906).

Ghosh, S., Laksana, E., Scherer, S., & Morency, L.-P. (2015) A multi-label convolutional neural network approach to cross-domain action unit detection. In *ACII*.

Greenwald, M. K., Cook, E. W., & Lang, P. J. (1989). Affective judgment and psychophysiological response: Dimensional covariation in the evaluation of pictorial stimuli. *Journal of Psychophysiology*, *3*(1), 51–64.

Hayat, M., Bennamoun, M., & El-Sallam, A. A. (2012). Evaluation of spatiotemporal detectors and descriptors for facial expression recognition. In *International Conference on Human-System Interaction* (pp. 43–47).

Huang, X., Zhao, G., Zheng, W., & Pietikäinen, M. (2012). Towards a dynamic expression recognition system under facial occlusion. *Pattern Recognition Letters*, *33*(16), 2181–2191.

Jeni, L., Cohn, J. F, & Kanade, J. F. (2015). Dense 3d face alignment from 2d videos in real-time. In *FG*.

Jiang, B., Valstar, M. F, & Pantic, M. (2011). Action unit detection using sparse appearance descriptors in space-time video volumes. In *FG* (pp. 314–321).

Jolliffe, I. (2002). *Principal component analysis*. NewYork: Wiley.

Kotsia, I., Buciu, I., & Pitas, I. (2008). An analysis of facial expression recognition under partial facial image occlusion. *Image and Vision Computing*, *26*(7), 1052–1067.

Linusson, H. (2013). *Multi-output random forests*. University of Borås/School of Business and IT.

Liu, M., Li, S., Shan, Shiguang, S., & Chen, X. (2013). Enhancing expression recognition in the wild with unlabeled reference data. In *ACCV* (pp. 577–588).

Liu, M., Li, S., Shan, S., & Chen, X. (2015). Au-inspired deep networks for facial expression feature learning. *Neurocomputing*, *159*, 126–136.

Lucey, P., Cohn J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The extended cohn-kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *CVPR Workshops* (pp. 94–101).

Mavadati, S. M., Mahoor, M. H., Bartlett, K., Trinh, P., & Cohn, J. F. (2013). DISFA: A spontaneous facial action intensity database. *Transactions on Affective Computing*, *4*(2), 151–160.

Nicolle, J., Bailly, K., & Chetouani, M. (2015). Facial action unit intensity prediction via hard multi-task metric learning for kernel regression. In *FG*.

Pei, Y., Kim, T.-K., & Zha, H. (2013). Unsupervised random forest manifold alignment for lipreading. In *ICCV* (pp. 129–136).

Ranzato, M. A., Susskind, J., Mnih, V., & Hinton, G. (2011). On deep generative models with applications to recognition. In *CVPR* (pp. 2857–2864).

Ren, S., Cao, X., Wei, Y., & Sun, J. (2014). Face alignment at 3000 fps via regressing local binary features. In *CVPR* (pp. 1685–1692).

Rifai, S., Bengio, Y., Courville, A., Vincent, P., & Mirza, M. (2012). Disentangling factors of variation for facial expression recognition. In *ECCV*.

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. In *ICML* (pp. 833–840).

Sandbach, G., Zafeiriou, S., Pantic, M., & Rueck, D. (2011). A dynamic approach to the recognition of 3D facial expressions and their temporal models. In *FG* (pp. 406–413).

Savran, A., Alyüz, N., Dibeklioğlu, H., Çeliktutan, O., Gökberk, B., Sankur, B., & Akarun, L. (2008). Bosphorus database for 3d face analysis. In *Biometrics and Identity Management* (pp. 47–56).

Sénéchal, T., Rapp, V., Salam, H., Seguier, R., Bailly, K., & Prevost, L. (2012). Facial action recognition combining heterogeneous features via multikernel learning. *TSMC-B* (pp. 42).

Shan, C., Gong, S., & McOwan, P. W. (2009). Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, *27*(6), 803–816.

Sun, Y., & Yin, L. (2008). Facial expression recognition based on 3D dynamic range model sequences. In *ECCV* (pp. 58–71).

Van de Weijer, J., Ruiz, A., & Binefa, X. (2015). From emotions to action units with hidden and semi-hidden-task learning. In *ICCV*.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, *11*, 3371–3408.

Wallhoff, F. (2006). Database with facial expressions and emotions from technical university of Munich (feedtum).

Xiong, X., & De la Torre, F. (2013). Supervised descent method and its applications to face alignment. In *CVPR* (pp. 532–539).

Xu, L., & Mordohai, P. (2010). Automatic facial expression recognition using bags of motion words. In *BMVC* (pp. 1–13).

Yin, L., Chen, X., & Sun, Y. (2008). Tony Worm, and Michael Reale. A high-resolution 3D dynamic facial expression database. In *FG* (pp. 1–6).

Zeng, Z., Pantic, M., Roisman, G. I., & Huang, T. S. (2009). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(1), 39–58.

Zhang, L., Tjondronegoro, D., & Chandran, V. (2014). Random Gabor based templates for facial expression recognition in images with facial occlusion. *Neurocomputing*, *145*, 451–464.

Zhang, X., Yin, L., Cohn, J. F., Canavan, S., Reale, M., Horowitz, A., et al. (2014). BP4D-spontaneous a high-resolution spontaneous 3D dynamic facial expression database. *Image and Vision Computing*, *32*(10), 692–706.

Zhao, K., Chu, W.-S., De la Torre, F., Jeffrey, F. C., & Honggang, Z. (2015). Joint patch and multi-label learning for facial action unit detection. In *CVPR*.

Zhao, K., Chu, W.-S., & Zhang, H. (2016). Deep region and multi-label learning for facial action unit detection. In *CVPR*.

Zhao, X., Kim, T. K., & Luo, W. (2014). Unified face analysis by iterative multi-output random forests. In *CVPR* (pp. 1765–1772).

Zhong, L., Liu, Q., Yang, P., Liu, B., Huang, J., & Metaxas, D. N. (2012). Learning active facial patches for expression analysis. In *CVPR* (pp. 2562–2569).