Dynamic pose-robust facial expression recognition by multi-view pairwise conditional random forests

Arnaud Dapogny, Kevin Bailly, Séverine Dubuisson

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

Dynamic Pose-Robust Facial Expression Recognition by Multi-View Pairwise Conditional Random Forests

Arnaud Dapogny[1] and Kevin Bailly[1] and Séverine Dubuisson[1]
[1] Sorbonne Universités, UPMC Univ Paris 06
CNRS, UMR 7222, F-75005, Paris, France

Abstract—Automatic facial expression classification (FER) from videos is a critical problem for the development of intelligent human-computer interaction systems. Still, it is a challenging problem that involves capturing high-dimensional spatio-temporal patterns describing the variation of one's appearance over time. Such representation undergoes great variability of the facial morphology and environmental factors as well as head pose variations. In this paper, we use Conditional Random Forests to capture low-level expression transition patterns. More specifically, heterogeneous derivative features (e.g. feature point movements or texture variations) are evaluated upon pairs of images. When testing on a video frame, pairs are created between this current frame and previous ones and predictions for each previous frame are used to draw trees from Pairwise Conditional Random Forests (PCRF) whose pairwise outputs are averaged over time to produce robust estimates. Moreover, PCRF collections can also be conditioned on head pose estimation for multi-view dynamic FER. As such, our approach appears as a natural extension of Random Forests for learning spatio-temporal patterns, potentially from multiple viewpoints. Experiments on popular datasets show that our method leads to significant improvements over standard Random Forests as well as state-of-the-art approaches on several scenarios, including a novel multi-view video corpus generated from a publicly available database.

Index Terms—Spontaneous facial expression recognition, Dynamics, Transition classification, Conditional Random Forest, Decision Tree, Video, Multi-view, Pose-robust, Real-time.

INTRODUCTION

Over the last decades, automatic facial expression recognition (FER) has attracted an increasing attention [1], [2], as it is a fundamental step of many applications such as human-computer interaction, or assistive healthcare technologies. The rationale behind those works is that decrypting facial expressions can serve as an unobtrusive way of analyzing one's underlying emotional state. Towards this end, a rich background literature has been developed by the psychological community in order to define models that can accurately and exhaustively represent facial expressions.

One of the most long-standing and widely used model is the discrete categorization proposed in the cross-cultural studies conducted by Ekman [3], which introduced six basic expressions that are universally recognized: *happiness*, *anger*, *sadness*, *fear*, *disgust* and *surprise*. This has been used to build as an underlying expression model for most attempts for *prototypical* expression benchmarking and recognition scenarios [4], [5], [6], as the annotation process is quite intuitive. It can however show limitations for dealing with spontaneous expressions [7], as many of our daily affective behaviors are not covered by such prototypical emotions.

Another popular approach is the continuous dimensional representation of affect [8], which consists in describing expressions in terms of a small number of latent variables rather than discrete categorical attributes. Perhaps one of the most widely used model is the valence/activation (relaxed *vs.* aroused)/power (feeling of control)/expectancy (anticipation) model. This model is often further simplified as a $2D$ valence-activation representation. However, the projection of complex emotional states into such a low-dimensional embedding may result in loss of information. As a consequence, some expressions such as *surprise* cannot be represented correctly whereas some others can not be separated efficiently (*fear vs. anger*). Finally, the annotation process is less intuitive than with the categorical representation.

An alternative representation of facial expressions has been proposed under the form of the Facial Action Coding System (FACS) [9]. Here, facial expressions are decomposed as a combination of 44 facial muscle activations called Facial Action Units (AUs). AUs provide an intermediate face representation that is independent from interpretation and can in theory be combined in accordance with the so-called Emotional FACS (EMFACS) rules to describe any prototypical or spontaneous expression display. Unfortunately, the main drawback of this approach is that FACS-coding is generally cumbersome, and raters generally have to be highly trained, thus limiting the quantity of available data.

For those reasons, in this work we focus on categorical facial expression classification. However, there is nothing in our method that would prevent us from adapting our code to either the dimensional or FACS model, given appropriate data. More specifically, we aim at designing a FER system that:

- is able to reliably distinguish subtle expressions (e.g. *anger* or *sadness*). Because using dynamics of the expression helps disentangle the factors of variation [10], such system needs to exploit the temporal variations in videos rather than trying to perform recognition on still images. For that matter, we focus on combining the benefits of frame-based and dynamic classifications;

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

- is robust to contextual factors (e.g. lighting conditions) and can perform recognition from arbitrary viewpoints, depending on head pose variation or camera position;
- can be learned from available data and work in real-time on a standard computer with any basic webcam plugged in. Particularly, we do not use high-resolution $3D$ face scans because many approaches [11], [12] working on such data seem to perform poorly when applied on consumer sensors such as the Kinect. In addition, depth information may be unavailable in many applicative scenarios.

1 Related work

In this section we review recent works addressing FER from video. On the one's hand, recent approaches for FER from a frontal view can be divided in frame-based systems and dynamic ones. On the other hand, multi-view FER is generally performed statically.

1.1 Frame-based FER

The first category of FER systems are the so-called frame-based classifiers. For instance, Khan *et al.* [13] propose a human vision-inspired framework that applies classification upon Pyramid Histogram of Orientation Gradients (PHOG) features from salient facial regions. Happy *et al.* [14] extract prominent facial patches from the position of facial landmarks. A subset of discriminative salient patches can then be used for FER.

This category of approaches typically aims at outputting an expression prediction for each separate frame. Hence, they can generally be applied to classify each frame of a video without pre-segmentation. Unfortunately, they also suffer from a number of drawbacks. First, they typically require frame-level annotations for training, which can be a time-consuming process. Secondly, frame-level approaches essentially ignore a part of the information as they do not exploit the temporal evolution of the features. They also do not use the temporal correlations at the semantic level (e.g. is it plausible to predict *sadness* immediately after having recognised *happiness*?). Recently, Meguid *et al.* [15] obtained promising results by accumulating hybrid RF/SVM predictions into histograms computed using a sliding window.

In order to disentangle facial morphology from expression, other approaches explicitly normalize each image w.r.t. a neutral face representation. Mohammadi *et al.* [16] use a constrained smoothed l_0-norm sparse decomposition to infer facial expressions from differences of face images. However, the neutral face has to be provided beforehand, limiting the applicability of these methods. In order to circumvent this issue, the so-called dynamic FER methods typically make use of spatio-temporal information.

1.2 Dynamic FER

Dynamic information of facial expressions can be used in several ways: (a) at the feature-level, by using spatio-temporal image descriptors, and/or (b) at the semantic level, by modelling relationships between expressions or between successive phases (*onset*, *apex* and *offset*) of facial events. Generally speaking, effectively extracting suitable representations from spatio-temporal video patterns is a challenging problem as expressions may occur with various offsets and at different paces. There is no consensus either on how to combine those representations flexibly enough so as to generalize on unseen data and possibly unseen temporal variations. Common approaches employ spatio-temporal descriptors defined on fixed-size windows, optionally at multiple resolutions. Examples of such features include the so-called LBP-TOP [17], [18] and HOG3D [19] descriptors, which are spatio-temporal extensions of LBP and HOG features respectively. Authors in [20] use histograms of local phase and orientations. However, those kind of representations may lack the capacity to generalize to facial events that differ from training data on the temporal axis.

Approaches trying to address (b) aim at establishing relationships between high-level features and a sequence of latent states. Wang *et al.* [21] integrate temporal interval algebra into a Bayesian network to capture complex relationships among facial muscles. Sikka *et al.* [22] propose a novel latent ordinal model that allows weakly supervised learning. Such approaches generally require explicit dimensionality reduction techniques such as PCA or k-means clustering for training. In addition, training at the sequence level reduces the quantity of available training and testing data as compared to frame-based approaches, as there is only one expression label per video. Finally, these approaches require continuity of the sequences for both training and testing, and may lack the flexibility to handle failure cases.

In a previous work [23], we trained Random Forests upon pairs of images representing expression transition patterns. Those forests were conditioned on the expression label of the first frame to help reducing the variability. Although we obtained promising results for dynamic FER from frontal views of the videos, the proposed approach did not handle head pose variations.

1.3 Multi-view FER

Many approaches for multi-view FER consist in training a single classifier to describe every viewpoint. Zheng *et al.* [24] introduce a regional covariance matrix representation of face images to infer static facial expressions on a corpus constructed from the BU-3DFE database [5] with 35 different head poses up to ± 45 yaw and ± 30 pitch. Tariq *et al.* [25] address the same problem by using a translation invariant sparse coding of dense SIFT features. Eleftheriadis *et al.* [26] employ discriminative shared Gaussian processes to implicitly exploit the redundancy between multiple views of the same expressive images. However, such approach can struggle to capture the variability of the facial expressions when the number of training samples becomes important.

Alternatively, it is possible to learn a projection of a non-frontal views of a face image on a frontal one. Recently, Vieriu *et al.* [11] proposed to project $3D$ data of the face onto a head pose-invariant $2D$ representation. The visible fraction of the projected face is then used within a voting scheme to decipher the expression. FER can thus be performed using an off-the-shelf algorithm. In addition, the authors were able to perform FER under a broader range of poses, up to ± 90 yaw and ± 60 pitch. However, the proposed method requires high-resolution $3D$ face data that may not necessarily be available in multiple human-computer interaction scenarios, for instance when using images acquired with consumer sensors such as the Kinect.

Last but not least, some other works choose to learn one specific classifier per face view. During testing, the head pose is first estimated, then the best pose-specific expression classifier is applied. For instance, Moore *et al.* learn multi-class SVMs upon LBP features for multiple viewpoints. Such approaches offer several advantages over the previous ones: first, learning classifiers upon separate and more homogeneous face view data allows to considerably reduce the variability. As a consequence the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

classifiers can, in theory, more efficiently capture the subtle facial deformations between the expressions. Secondly, the runtime is the same as in the case of a single frontal view classifier, which may be a critical point for systems that try to project a given view on a frontal one. Finally, splitting the training data offers the advantage to reduce the memory usage, which can be important for learning on large databases. Those methods also face some impediments, such as the fact that (a) they require a reliable facial landmark alignment and head pose estimation, and (b) it implies dividing the data into several subsets. Nevertheless, (a) is barely a problem given that recent advances [27], [28], [29] for face alignment provide excellent results for head poses up to ± 45 yaw and ± 30 pitch, which is sufficient for most human-computer applications. Furthermore, (b) can be circumvented by the use of $3D$ face scans [6] from which we can generate a large corpus of videos for training multi-view dynamic classifiers.

2 Overview of Proposed Approach

In this paper, we introduce the Multi-View Pairwise Conditional Random Forest (MVPCRF) algorithm, which is a new formulation for training trees using low-level heterogeneous static (spatial) and dynamic (spatio-temporal derivative) features within the Random Forest (RF) framework. Conditional Random Forests have recently been used by Dantone *et al.* [30], Yang *et al.* [31] as well as Sun *et al.* [32] in the field of facial alignment and human pose estimation, respectively. They generated collections of trees for specific, quantized values of a global variable (such as head pose [30] and body torso orientation [32]) and used prediction on this global variable to draw dedicated trees, resulting in more accurate predictions. As depicted on Figure 1, we propose to condition pairwise trees on specific expression labels to reduce the variability of ongoing expression transitions from the first frame of the pair to the other one. Furthermore, similarly to [30], we can further condition the pairwise trees on a head pose estimation to add robustness towards head pose variations. When evaluating a video frame, each previous frame of the sequence is associated with this current frame to give rise to a pair. Pairwise trees are thus drawn from the dedicated PCRFs w.r.t. prediction for the previous frame. *In Extenso*, a head pose estimate can be used to draw trees from MVPCRFs for pose-robust FER. Finally, predictions outputted for each pair are averaged over time to produce a robust prediction for the current frame. Contributions of this work are listed below.

- A method for training pairwise random trees upon high-dimensional heterogeneous static and spatio-temporal derivative feature templates, with a conditional formulation that reduces the variability of the transition patterns.
- An extension of the traditional RF model averaging, that consists in averaging over time pairwise predictions to flexibly handle temporal variations.
- A method for performing multi-view dynamic FER that consists in further conditioning pairwise trees on a pose estimate. A tree sampling probability distribution is constructed from the data to allow a continuous shift between the pose-specific PCRF models.
- A new multi-view video corpus, that includes a method for aligning facial feature points on non-frontal sequences using an off-the-shelf feature point tracker. We provide source code for generating the data from $3D$ models using an available database [6].
- A complete PCRF framework that performs fully automatic dynamic FER with a multi-view extension, that can work on low-power engines thanks to an efficient implementation using integral feature channels.

The rest of the paper is organized as follows: in Section 3 we describe our adaptation of the RF framework to learn expression patterns on still images from high-dimensional, heterogeneous (geometric/appearance) features. In Section 4 we present the MVPCRF framework for capturing spatio-temporal patterns that represent facial expressions from multiple viewpoints. In Section 5 we explain how we generate a multi-view dynamic database for training and testing the models. In Section 6 we show how our PCRF algorithm improves the accuracy on several FER datasets compared to a static approach as well as state-of-the-art approaches. In Section 6.3 we report results from frontal view FER and in Section 6.4 we report accuracy for non frontal head poses and FER in the wild, showing that our formulation substantially increases the robustness to pose variations. In Section 6.5 we report the ability of our framework to run in real-time. Finally, we give concluding remarks on MVPCRF for FER and discuss upcoming perspectives.

3 Random Forests for FER

3.1 Random Forests

Random Forests (RFs) is a popular learning framework introduced in the seminal work of Breiman [33]. They have been used to a significant extent in computer vision and for FER tasks in particular due to their ability to handle high-dimensional data such as images or videos as well as being naturally suited for multiclass classification tasks. They combine random subspace and bagging methods to provide performances similar to the most popular machine learning methods, such as SVM or neural networks.

RFs are classically built from the combination of T decision trees grown from bootstraps sampled from the training dataset. In our implementation, we downsample the majority classes within the bootstraps in order to enforce class balance. 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.

Individual trees are grown using a greedy procedure that involves, for each node, the measure of an impurity criterion $H_{\phi,\theta}$ (which is traditionally either defined as the Shannon entropy or the Gini impurity measurement) relatively to a partition of the images x with label $l \in \mathcal{L}$, that is induced by candidate binary split functions $\{\phi, \theta\} \in \Phi$. More specifically, we use multiple parametric feature templates to generate multiple heterogeneous split functions, that are associated with a number of thresholds θ. In what follows, by abuse of notations we will refer to $\phi^{(i)}$ as the i^{th} feature template and $k^{(i)}$ as the number of candidates generated from this template. The "best" binary feature among all features from the different templates (*i.e.* the one that minimizes the impurity criterion $H_{\phi,\theta}$) is set to produce a data split for the current node. Then, those steps are recursively applied for the left and right subtrees with accordingly rooted data until the label distribution at each node is homogeneous, where a leaf node is set. This procedure for growing trees is summarized in Algorithm 1.

During evaluation, an image x is successively rooted left or right of a specific tree t according to the outputs of the binary

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

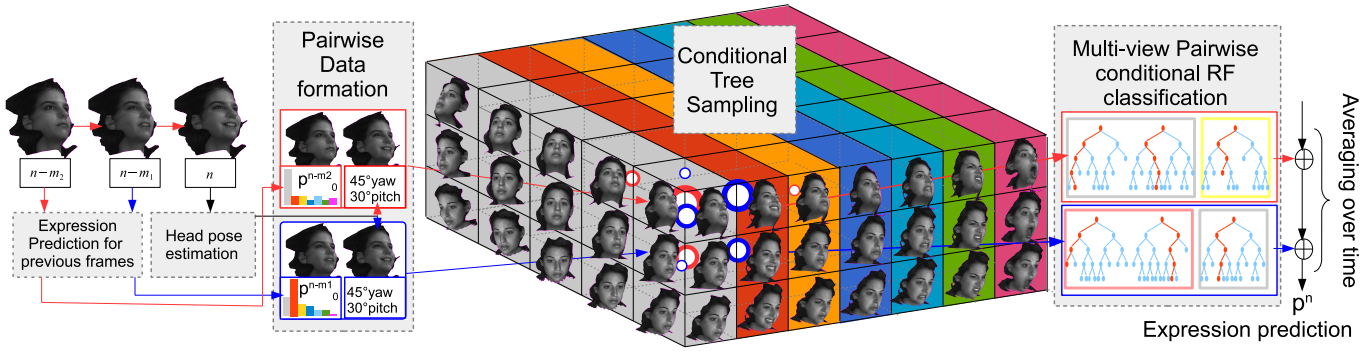


Fig. 1. Flowchart of our MVPCRF FER method. When evaluating a video frame indexed by n, pairs are created between this current frame and previous frames $n-m_1, n-m_2, ...$ Randomized trees trained upon a pairwise dataset are then drawn conditionally to head pose estimation as well as expression probabilities for the previous frames. Finally, predictions outputted for each pair are averaged over time to give rise to an expression probability p^n for the current frame. This prediction is used as a tree sampling distribution for classifying the following frames. Best viewed in color.

Algorithm 1 Tree Growing algorithm `treeGrowing`

input: images x with labels l, root node n, number of candidate features $\{k^{(i)}\}_{i=1,2,3}$ for templates $\{\phi^{(i)}\}_{i=1,2,3}$

if image labels are homogeneous with value l_0 **then**
  set node as terminal, with probabilities p_t to 1 for l_0, 0 elsewhere
else
  generate an empty set of split candidates Φ
  for all feature templates i **do**,
    generate a set $\Phi^{(i)}$ of $k^{(i)}$ candidates $\{\phi^{(i)}, \theta\}$
    $\Phi \leftarrow \Phi \cup \Phi^{(i)}$
  end for
  for $\{\phi, \theta\} \in \Phi$ **do**
    compute the impurity criterion $H_{\phi,\theta}(x)$
  end for
  split data w.r.t. $\arg\min_{\{\phi,\theta\}}\{H_{\phi,\theta}(x)\}$ in left and right subsets x_l and x_r
  create left (n_l) and right (n_r) children of node n
  call `treeGrowing`(x_l,n_l,$\{k^{(i)}\}_{i=1,2,3}$)
  call `treeGrowing`(x_r,n_r,$\{k^{(i)}\}_{i=1,2,3}$)
end if

tests, until it reaches a leaf node. The tree thus returns a probability $p_t(l|x)$ which is set to either 1 for the represented class, or to 0. Prediction probabilities are then averaged among the T trees of the forest (Equation (1)).

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

Note that the robustness of the RF prediction framework comes from (a) the strength of individual trees and (b) the decorrelation between those trees. By growing trees from different bootstraps of available data and with the random subspace algorithm (e.g. examining only a subset of features for splitting each node) we generate weaker, but less correlated trees that provide better combination predictions than CART or C4.5 procedures [34].

3.2 Heterogeneous feature templates

Feature templates $\phi^{(i)}$ include both geometric (*i.e.* computed from previously aligned facial feature points) and appearance features. Each of these templates have different input parameters that are randomly generated during training by uniform sampling over their respective variation range. Also, during training, features are generated along with a set of candidate thresholds θ to produce binary split candidates. For each template $\phi^{(i)}$, the upper and lower bounds are estimated from the training data and candidate thresholds are drawn from uniform distributions within this range.

We use two different geometric feature templates which are generated from the set of facial feature points $f(x)$ aligned on image x with SDM [27]. The first geometric feature template $\phi^{(1)}_{a,b}$ is the distance between feature points f_a and f_b, normalized w.r.t. inter-ocular distance $iod(f)$ for scale invariance (Equation 2).

$$\phi^{(1)}_{a,b}(x) = \frac{||f_a - f_b||_2}{iod(f)} \tag{2}$$

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, we use the cosine and sine instead of the raw angle. Thus, $\phi^{(2)}$ outputs either the cosine or sine of angle $\widehat{f_a f_b f_c}$, depending on the value of the boolean parameter λ (Equation (3)):

$$\phi^{(2)}_{a,b,c,\lambda}(x) = \lambda\cos(\widehat{f_a f_b f_c}) + (1-\lambda)\sin(\widehat{f_a f_b f_c}) \tag{3}$$

We use Histogram of Oriented Gradients (HOG) as our appearance features for their descriptive power and robustness to global illumination changes. In order to ensure fast feature extraction, we use integral feature channels as introduced in [35]. First, images are rescaled to a constant size of 250×250 pixels. Then, we compute horizontal and vertical gradients on the image and use these to generate 9 feature maps, the first one containing 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. Thus, we define the appearance feature template $\phi^{(3)}_{\tau,ch,s,\alpha,\beta,\gamma}$ as an integral histogram computed over channel ch within a window of size s normalized w.r.t. the inter-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 $f(x)$. Also, we store the gradient magnitude in the first channel to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

normalize the histograms. Thus, HOG features can be computed with only 4 access to the integral channels (plus normalization).

However, the proposed static RF does not use the dynamics of the expressions, which is the purpose of the next section.

4 Learning Temporal Patterns from Multiple Viewpoints

4.1 Learning PCRF with heterogeneous derivative feature templates

In this section we now consider pairs of images (x',x) to train trees t that aim at outputting probabilities $p_t(l|x',x,l')$ of observing label $l(x)=l$ given image x' and subject to $l(x')=l'$, as shown in Figure 2. More specifically, for each tree t among the T trees of a RF dedicated to transitions starting from expression label l', we randomly draw a fraction of subjects $\tilde{\mathcal{S}} \subset \mathcal{S}$. Then, for each subject $s \in \tilde{\mathcal{S}}$ we randomly draw images x'_s that specifically have label l'. We also draw images x_s of every label l and create the pairs (x'_s, x_s) with label l. Note that the two images of a pair need to belong to the same subject, but not necessarily to the same video. Indeed, we create pairs from images sampled across different sequences for each subject to cover all sorts of ongoing transitions. We then balance the pairwise bootstrap by downsampling the majority class w.r.t. the pairwise labels. Eventually, we construct tree t by calling Algorithm 1. Those steps are summarized in Algorithm 2.

Algorithm 2 Training a PCRF

input: images x with labels l, number of candidate features $\{k^{(i)}\}_{i=1,\dots,6}$ for templates $\{\phi^{(i)}\}_{i=1,\dots,6}$

```
for all l' ∈ L do
    for t = 1 to T do
        randomly draw a fraction S̃ ⊂ S of subjects
        pairs ← {}
        for all s ∈ S̃ do
            draw samples x'_s with label l'
            draw samples x_s for each label l
            create pairwise data (x'_s, x_s) with label l
            add element (x'_s, x_s) to pairs
        end for
        balance bootstrap pairs with downsampling
        create new root node n
        call treeGrowing(pairs,n,{k^(i)}_{i=1,...,6})
    end for
end for
```

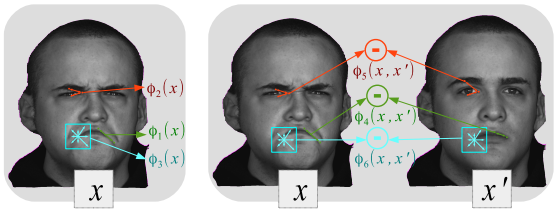


Fig. 3. Static (left) and pairwise (right) feature templates.

As shown on Figure 3, candidates for splitting the nodes are generated from an extended set of 6 feature templates $\{\phi^{(i)}\}_{i=1,\dots,6}$, three of which being the static features described in Section 3, that are applied to the second image x of the pair (x', x), for which we want to predict facial expressions. The three remaining feature templates are dynamic features defined as the derivatives of static templates $\phi^{(1)}$, $\phi^{(2)}$, $\phi^{(3)}$ with the exact same parameters. Namely, we have:

$$\begin{cases} \phi^{(1)}_{a,b}(x',x) & = \phi^{(1)}_{a,b}(x) \\ \phi^{(2)}_{a,b,c,\lambda}(x',x) & = \phi^{(2)}_{a,b,c,\lambda}(x) \\ \phi^{(3)}_{\tau,ch,s,\alpha,\beta,\gamma}(x',x) & = \phi^{(3)}_{\tau,ch,s,\alpha,\beta,\gamma}(x) \\ \phi^{(4)}_{a,b}(x',x) & = \phi^{(1)}_{a,b}(x) - \phi^{(1)}_{a,b}(x') \\ \phi^{(5)}_{a,b,c,\lambda}(x',x) & = \phi^{(2)}_{a,b,c,\lambda}(x) - \phi^{(2)}_{a,b,c,\lambda}(x') \\ \phi^{(6)}_{\tau,ch,s,\alpha,\beta,\gamma}(x',x) & = \phi^{(3)}_{\tau,ch,s,\alpha,\beta,\gamma}(x) - \phi^{(3)}_{\tau,ch,s,\alpha,\beta,\gamma}(x') \end{cases} \tag{4}$$

As in Section 3, thresholds for the derivative features $\phi^{(4)}$, $\phi^{(5)}$, $\phi^{(6)}$ are drawn from uniform distributions with new dynamic template-specific ranges estimated from the pairwise dataset.

Note that, as compared to a static RF, a PCRF model is extended with new derivative features that are estimated from a pair of images. When applied on a video, predictions for several pairs are averaged over time in order to produce robust estimates of the probability predictions.

4.2 Model averaging over time

We denote by $p^n(l)$ the prediction probability of label l for a video frame x^n. For a purely static RF classifier this probability is given by Equation (5):

$$p^n(l) = \frac{1}{T}\sum_{t=1}^{T} p_t(l|x^n) \tag{5}$$

In order to use spatio-temporal information, we apply pairwise RF models to pairs of images (x^m, x^n) with $\{x^m\}_{m=n-1,\dots,n-N}$ the previous frames in the video. Those pairwise predictions are averaged over time to provide a new probability estimate p^n that takes into account past observations up to frame n. Thus, if we do not have prior information for those frames the probability p^n becomes:

$$p^n(l) = \frac{1}{NT}\sum_{m=n-N}^{n-1}\sum_{t=1}^{T} p_t(l|x^m, x^n) \tag{6}$$

In what follows, Equation (5) and Equation (6) will respectively be referred to as the *static* and *full models*. Trees from the full model are likely to be stronger that those of the static one since they are grown upon an extended set of features. Likewise, the correlation between the individual trees is also lower thanks to the new features as well as the averaging over time. However, spatio-temporal information can theoretically not add much to the accuracy if the variability of the pairwise data points is too large.

In order to decrease this variability, we assume that there exists a probability distribution $p_0^m(l')$ to observe the expression label l' at frame m. Note that those probabilities can be set to purely static estimates (which is necessarily the case for the first video frames) or dynamic predictions estimated from previous frames. A comparison between those approaches can be found in Section 6.3. In such a case, for frame m, pairwise trees are drawn from the trees collections (each one being conditioned to one expression label for the first frame of the pair) by sampling the distribution p_0^m,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

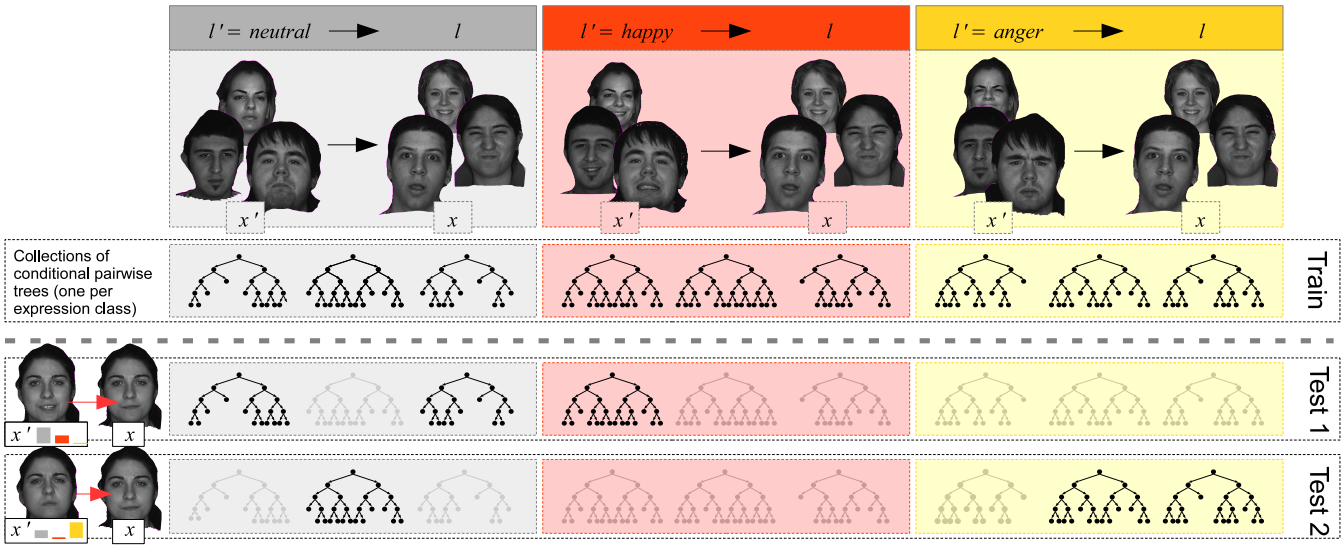


Fig. 2. Expression recognition from pairs of images using PCRF. Expression probability predictions of previous images are used to sample trees from dedicated pairwise tree collections (one per expression class) that are trained using subsets of the (pairwise) training dataset, with only examples of ongoing transitions from a specific expression towards all classes. The resulting forest thus outputs an expression probability for a specific pair of images.

as shown in Figure 2. More specifically, for each expression label l' we randomly select $\mathcal{N}(l')$ trees over a PCRF model dedicated to transitions that start from expression label l', trained with the procedure described in Section 4.1. Equation (6) thus becomes:

$$p^n(l) = \frac{1}{NT} \sum_{m=n-N}^{n-1} \sum_{l' \in \mathcal{L}} \sum_{t=1}^{\mathcal{N}(l')} p_t(l|x^m, x^n, l') \qquad (7)$$

Where $\mathcal{N}(l') \approx T p_0^m(l')$ and $T = \sum_{l' \in \mathcal{L}} \mathcal{N}(l')$ are the number of trees dedicated to the classification of each transition, which can be set in accordance with CPU availability. In our experiments, we will refer to Equation (7) as the *conditional model*. This conditional formulation helps to reduce the variability of the derivative features for each specialized pairwise RF. When predicting expression for a frame of a video, we can effectively use robust sequence-level expression estimates by averaging over time predictions conditioned on multiple, independent previous frames. Section 6 shows that using PCRF models for FER leads to significant improvements over both static and full models.

4.3 Multi-view formulation

In order to design a pose-robust recognition framework, we propose to condition the models w.r.t a head pose estimate $\omega(x^n)$ for frame n. For that matter we quantize the pose space Ω in $k = \Gamma \times B$ pose bins $\{\Omega_i = \Omega_{\gamma_i, \beta_i}\}_{i=1,\dots,k}$, that are defined around yaw and pitch angles γ_i and β_i, respectively. We can thus rewrite Equation 5 as a static multi-view model (MVRF):

$$p^n(l) = \frac{1}{T} \sum_{\Omega_i \in \Omega} \sum_{t=1}^{\mathcal{N}(\Omega_i)} p_t(l|x^n, \Omega_i) \qquad (8)$$

At frame n, the head pose $\omega(x^n)$ is estimated first using an off-the-shelf posit algorithm [36]. Then, for each pose bin Ω_i, a number $\mathcal{N}(\Omega_i)$ of trees are selected based on a pose sampling probability distribution $\mathcal{P}_{\Omega_i}(\omega^n)$ that we construct from the training data repartition, as it will be explained in Section 5. Furthermore, we adapt Equation (7) by conditioning the expression-conditional model on pose estimation $\omega(x^n)$ (Equation (9)):

$$p^n(l) = \frac{1}{T} \sum_{m=n-N}^{n-1} \sum_{\Omega_i \in \Omega} \sum_{l' \in \mathcal{L}} \sum_{t=1}^{\mathcal{N}(l', \Omega_i)} p_t(l|x^n, x^m, \Omega_i, l') \qquad (9)$$

In what follows, we refer to this model as the *multi-view PCRF* (MVPCRF) model. In this formulation, for computing the pairwise probability between frames n and m, we first estimate the head pose for frame n. Then, for each pose bin Ω_i and expression label l', we select a number of trees equal to $\mathcal{N}(l', \Omega_i)$ (Equation (10)):

$$\mathcal{N}(l', \Omega_i) \approx T \mathcal{P}_{\Omega_i}(\omega(x^n)) p_0^m(l') \qquad (10)$$

The number of trees allocated to classify each transition is:

$$T = \sum_{\Omega_i \in \Omega} \sum_{l' \in \mathcal{L}} \mathcal{N}(l', \Omega_i) \qquad (11)$$

Note that the tree sampling distribution proposed in Equation 10 supposes that the head pose estimate do not vary that much between frames $n - N$ and n. Should that be the case, MVPCRF can be trained from pairs of images from different viewpoints. It also assumes the independence of head pose and expression prior, which is not problematic for training on posed expression data. However, such assumption may not hold for spontaneous datasets for which expressions as *surprise* or *fear* may involve specific head motion (e.g. recoil). In such case, prior conditionals may be estimated from the training corpus beforehand. Also, as stated in [30], [32] using conditional models usually involves one major pitfall, which lies in the reduction of the number of training examples used to train each separate classifier. This is barely a problem for the training of a PCRF model, as naturally many examples of each ongoing transition can be sampled from the datasets. Furthermore, for the MVPCRF model we can generate a new database that contains a large number of training examples for each pose bin using the high-resolution $3D$-models from the BU-4DFE database [6].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

Fig. 4. Boot process for multi-view data generation with aligned feature points

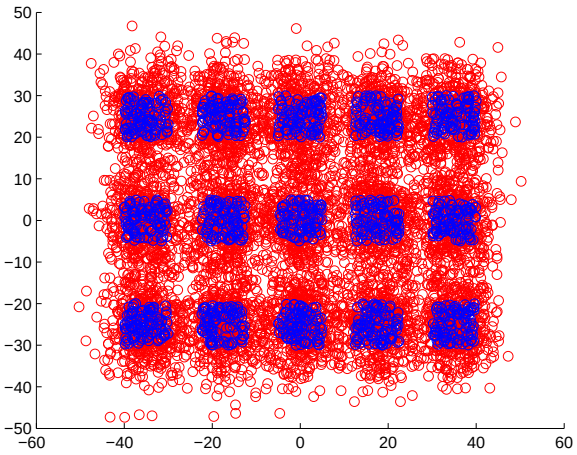


Fig. 5. Data repartition across the 15 generated pose bins. Blue circles: angles associated to the sequences (γ_i^s,β_j^s), red: individual frames

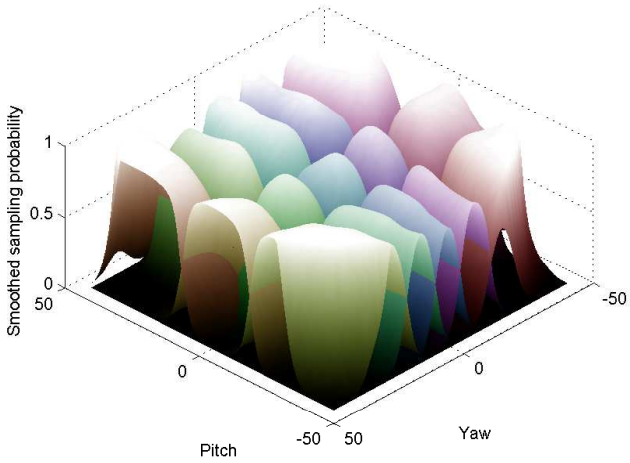


Fig. 6. Pose sampling probability distributions $\mathcal{P}_{\Omega_i}(\omega^n)$ constructed by smoothing the data repartition for each pose bin

5 Multi-view database generation

Each texture frame of the BU-4DFE database is associated with a high-resolution $3D$ VRML model containing approximately 35000 vertices, that we use to train our MVPCRF classifier as well as to design a new dataset for multi-view video FER. Many approaches [25], [11] present results for static multi-view FER using the BU-3DFE database [5]. To do that, for each static image, the authors typically render $3D$ meshes from a viewpoint with fixed yaw and pitch rotation angles. However, for video FER, head pose does not necessarily remain constant throughout a video. Furthermore, from the perspective of a fully automatic multi-view FER system, we typically aim at covering a specific head pose range rather than a discrete, arbitrary set of viewpoints. Hence, we propose to generate rotated versions of the videos by assigning each sequence a yaw-pitch variation from the frontal video. More specifically, our goal is to cover the same “useful” range as in [25], [11] (*i.e* ± 45 yaw, ± 30 pitch). We thus generate $k = 5 \times 3$ bins $\{\Omega_i = \Omega_{\gamma_i,\beta_i}\}_{i=1,\dots,k}$ with $\{\gamma_i\} = \{0, \pm 17.5, \pm 35\}$ and $\{\beta_i\} = \{0, \pm 25\}$ the mean rotation angles respectively in yaw and pitch. Each sequence s is thus associated with rotation angles:

$$\begin{cases} \gamma_i^s = \gamma_i + \gamma' \\ \beta_j^s = \beta_j + \beta' \end{cases} \tag{12}$$

Where γ' and β' are random variations uniformly drawn from the ranges $[-\sigma_\gamma, \sigma_\gamma]$ and $[-\sigma_\beta, \sigma_\beta]$, respectively. σ_γ and σ_β respectively denote the expected yaw and pitch width of the pose bins. In order to set those values, we measure the standard deviation of the head pose angles on the frontal view (3.6 and 5.9 in yaw and pitch respectively). We then set $\sigma_\gamma = \sigma_\beta = 5\,\text{deg}$ to allow a small overlap, thus a smoother interpolation between adjacent pose bins. The data distribution among the generated pose bins can be seen in Figure 5. For each frame of each sequence s, we generate 15 frames by rotating the camera (position, direction and up vector). We also turn off the camera headlight and add an ambient light node to the VRML virtual environment.

The next step is to align facial feature points on the rotated sequences. However, the standard pipeline of applying a frontal or full profile face detection before aligning the feature points from the output face rectangle is bound to fail when the yaw/pitch becomes important and only a few images can correctly be aligned. In order to circumvent those issues, we generate “boot” sequences using the first image of each video. Those sequences contain 20 frames and show a very progressive rotation of the first frame starting from a frontal view and ending on the expected viewpoint. We apply the OpenCV Viola-Jones face detector [37] on the first frame of the boot sequence (frontal view). Then we align facial feature points with the SDM tracker [27] on the retrieved face rectangle. Feature points are then tracked throughout the boot sequence. Once the *boot* is completed, feature points are tracked on all the frames of the rotated expression videos (Figure 4). Finally, we crop the facial images to a constant size based on the feature point location and generate a total of 906030 images.

Lastly, we construct our multi-view training set by manually selecting the neutral and apical frames using the same subsets as in the frontal case. Because precisely identifying such frames is difficult in the general case, we select 3 images before and after each peak frame for training to ensure robustness to small errors

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

in the localization of those apex frames. Also, in order to filter out the incorrectly aligned frames, we automatically discard the frames for which more than 5 feature points do not lie on the facial mesh. Our final training set thus consists of 122623 face images. Note however that we did not apply any manual check to remove the misaligned frames, or the ones for which the $3D$ models contain some distortions. The image generation process took about 5 days to complete on an *I7-4770* CPU on a `Matlab` environment. In order to ensure reproducibility of the results as well as to facilitate further research, we plan on releasing the peak frame annotation used for training the classifiers as well as the code for rendering the files and generating the boot and expression videos. For each of the retrieved frames, we use the posit algorithm [36] to estimate head pose from the feature points. Such setting allows to use the same head pose estimation for training and testing, as compared to, e.g. constructing the pose sampling distribution from the ground truth generated positions. Then, we compute the pose sampling probability distribution for each pose bin $\mathcal{P}_{\Omega_i}(\omega(x^n))$ by applying a Gaussian smoothing on the training data repartition in the yaw/pitch space (Figure 6). Thanks to the booting procedure discussed above, the number of training samples between the different pose bins is roughly equivalent. However, this might not be the case for other datasets, where constructing a sampling probability from the data offers the advantage to implicitly downweight the sampling of pose-specific trees relatively to the amount of training data.

6 Experiments

In this section, we report accuracies obtained on two different FER scenarios. In Section 6.3 we report comparisons between different classification models on two well-known frontal FER databases, the Extended Cohn-Kanade and BU-4DFE databases. Furthermore, in order to evaluate the capabilities of the learned models to generalize on spontaneous FER scenarios, we report classification results for cross-database evaluation on two spontaneous databases, namely the FG-NET FEED and BP4D databases. We highlight that our conditional formulation of dynamic integration increases the recognition accuracy on such difficult tasks. Furthermore, in Section 6.4 we also evaluate our approach on multi-view video FER as well as FER in the wild. Finally, in Section 6.5 we show the real-time capability of our system.

6.1 Databases

The CK+ or **Extended Cohn-Kanade database [4]** contains 123 subjects, each one associated with various numbers of expression records. Those records display a very gradual evolution from a *neutral* class towards one of the 6 universal facial expressions described by Ekman [3] (*anger*, *happiness*, *sadness*, *fear*, *digust* and *surprise*) plus the non-basic expression *contempt*. Expressions are acted with no head pose variation and their duration is about 20 frames. From this dataset we use 309 sequences, each one corresponding to one of the six basic expressions, and use the three first and last frames from these sequences for training. We did not include sequences labelled as *contempt* because CK+ contains too few subjects performing *contempt* and other expressions to train the pairwise classifiers.

The BU-4DFE database [6] contains 101 subjects, each one displaying 6 acted facial expressions with moderate head pose variations. Expressions are still prototypical but they are generally exhibited with much 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 annotations, we manually selected neutral and apex of expression frames. More specifically, we select 3 images before and after each peak frame, making a total of 8219 frames for training. Each frame is associated with high-resolution $3D$ model data recorded using a Di3D device, that we use in our experiments to generate expression videos from multiple viewpoints.

The BP4D database [7] contains 41 subjects. Each subject was asked to perform 8 tasks, each one supposed to give rise to one of the 8 spontaneous expressions (*anger*, *happiness*, *sadness*, *fear*, *digust*, *surprise*, *embarrassment* or *pain*). In [7] the authors extracted subsequences of about 20 seconds for manual annotation, since these subsets contain the most expressive behaviors.

The FG-NET FEED database [38] contains 19 subjects, each one recorded three times while performing 7 spontaneous expressions (the six universal expressions, plus the *neutral* one). The data contain low-intensity emotions, very short expression displays, as well as moderate head pose variations.

The AFEW database [39] has been collected from movies, displaying expressions in unconstrained conditions. As the test set is unreleased, we report accuracies on the validation set, which contains 410 videos, each containing 60 frames on average.

6.2 Experimental setup

7-class RF (static) and PCRF (full and conditional) models are trained on the CK+ and BU-4DFE datasets using the set of hyperparameters described in Table 1. Note however that extensive testing showed that the values of these hyperparameters had a very subtle 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. Nevertheless, we report those settings for reproducibility concerns. Also, for a fair comparison between static and pairwise models, we use the same total number of feature evaluations for generating the split nodes. For every test, we report results averaged over 5 different runs, with a standard deviation lower than 0.25% between each run.

TABLE 1
Hyperparameter settings

Hyperparameters	value(RF)	value(PCRF)
Nb. of $\phi^{(1)}$ features	40	20
Nb. of $\phi^{(2)}$ features	40	20
Nb. of $\phi^{(3)}$ features	160	80
Nb. of $\phi^{(4)}$ features	-	20
Nb. of $\phi^{(5)}$ features	-	20
Nb. of $\phi^{(6)}$ features	-	80
Data ratio per tree	$2/3$	$2/3$
Nb. of thresholds	25	25
Total nb. of features	6000	6000
Nb. of trees	500	500

During the evaluation, the prediction is initialized in a fully automatic way from the first frame using the static classifier. Then, for the full and conditional models, probabilities are estimated for each frame using transitions from previous frames only, bringing us closer to a real-world scenario. However, although it uses transitional features, our system is essentially a frame-based classifier that outputs an expression probability for each separate

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

video frame. This is different from, for example, a HMM, that aims at predicting a probability related to all the video frames. Thus, in order to evaluate our classifier on video FER tasks, we acknowledge correct classification if the maximum probability outputted for all frames corresponds to the ground truth label. This evaluates the capability of our system to retrieve the most important expression mode in a video, as well as the match between the retrieved mode and the ground truth label.

For the tests on CK+ and BU-4DFE databases, both static and transition classifiers are evaluated using the Out-Of-Bag (OOB) error estimate [33]. More specifically, bootstraps for individual trees of both static and pairwise classifiers are generated at the subject level. Thus, during evaluation, each tree is applied only on subjects that were not used for its training. The OOB error estimate is an unbiased estimate of the true generalization error [33] which is faster to compute than Leave-One-Subject-Out or k-fold cross-evaluation estimates. Also, it has been shown to be generally more pessimistic than traditional error estimates [40], further empathizing the quality of the proposed contributions.

6.3 FER from frontal view videos

6.3.1 FER on prototypical data

In order to validate our approach on frontal view videos, we compared our conditional model to a purely static model and a full model, for a variety of dynamic integration parameters (the length of the temporal window N and the step between those frames *Step*) on the BU-4DFE database. We also evaluated the interest of using a *dynamic* probability prediction for previous frames (*i.e.* the output of the pairwise classifier for those frames) versus a *static* one. Average results are provided in Figure 7. For CK+ database, sequences are generally too short to show significant differences when varying the temporal window size or the step size. Thus we only report accuracy for full and conditional models with a window size of 30 and a step of 1. Per-expression accuracies and F1-scores for both Cohn-Kanade and BU-4DFE databases are shown in Figure 8.

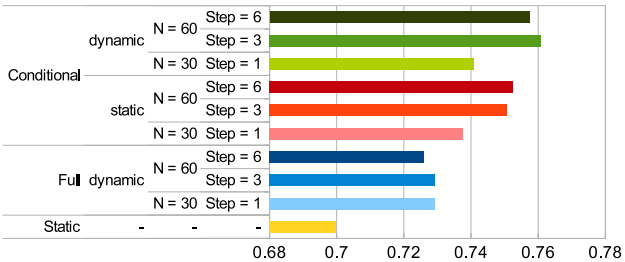


Fig. 7. Average accuracy rates obtained for various temporal integration parameters on the BU-4DFE database

Figure 8 reveals that facial expressions involving large deformations (e.g. $surprise$ and $happy$) are recognized with very high accuracies. $Disgust$ is also recognized quite well for both databases and for all the models. However, more subtle expressions such as $anger$ and $sadness$ rank among the lowest. For those expressions, the addition of spatio-temporal allows to increase the recognition accuracy as compared to a static RF model. As in many other works on facial expressions, accuracies for $fear$ are lower than for the other expressions, as it can be quite subtle in some cases where the eyes are open a little bit wider. Moreover, this expression also displays larger variability than the others on these databases. Overall, modelling transition patterns through PCRF allows to significantly increase the recognition accuracy as well as the balanced $F1$-score, for all expressions on both CK+ and BU-4DFE databases. We believe that this is due to the extra dynamic features that provide both robustness and decorrelation of the individual decision trees.

Figure 8 also shows that the conditional model outperforms the full model on both databases, which is probably due to the fact that using only a restricted set of ongoing expression transitions for training allows to better capture the variability of the spatio-temporal features for the dedicated pairwise forests. This is particularly true on the CK+ database, where the number of pairwise data points is not enough for the full model to capture the variability of all possible ongoing transitions, hence justifying the lower accuracy. Table 7 also shows that it is better to look backward for more frames in the sequence (N = 60) with less correlation between the frames (*Step* = 3 or 6). Again, such setting allows to take more decorrelated paths in the individual trees, giving a better recombination after averaging over time.

A compilation of comparisons to other state-of-the-art approaches for FER can be found in Table 2. On the CK+ dataset, we compare our algorithms with recent works reporting results on the same subset of sequences (*i.e.* not including *contempt*). Such comparisons are to be put into perspective as the evaluation protocols differ between the methods. Nevertheless, PCRF provides slightly better results than those reported in [16] ($+3.2\%$) as well as in [20] ($+1.9\%$), [14] ($+2.3\%$) and [41] ($+3.2\%$). Finally, Liu *et al.* [42] obtain slightly better results than ours ($+0.3\%$), but their method aim at classifying the last frame of the video specifically, whereas ours automatically retrieves the apex as the maximum probability image throughout the sequence.

Moreover, to the best of our knowledge, our approach gives the best results on the BU-4DFE database for automatic FER from videos using $2D$ information only. It provides better results than the dynamic $2D$ approach [43] ($+9.1\%$), as well as the LBP-TOP approach presented in [18] ($+4.5\%$). Recently, Meguid *et al.* [15] obtained satisfying results using an original hybrid RF/SVM system. They trained on the static BU-3DFE database [5] and employ a post-classification temporal integration scheme. However our PCRF method achieved a significantly higher accuracy ($+3\%$) which shows the benefits of using dynamic information at the feature level.

TABLE 2
Comparisons with state-of-the-art approaches on prototypical data. The first and second best methods are highlighted in red and blue, respectively.

CK+ database	Accuracy
Mohammadi *et al.* [16]	93.2
Happy *et al.* [14]	94.1
Shojaeilangari *et al.* [20]	94.5
Mollahosseini *et al.* [41]	93.2
Liu *et al.* [42]	**96.7**
This work, RF	93.2
This work, PCRF	**96.4**
BU-4DFE database	**Accuracy**
Sun *et al.* [43]	67.0
Hayat *et al.* [18]	71.6
Meguid *et al.* [15]	**73.1**
This work, RF	70.0
This work, PCRF	**76.1**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

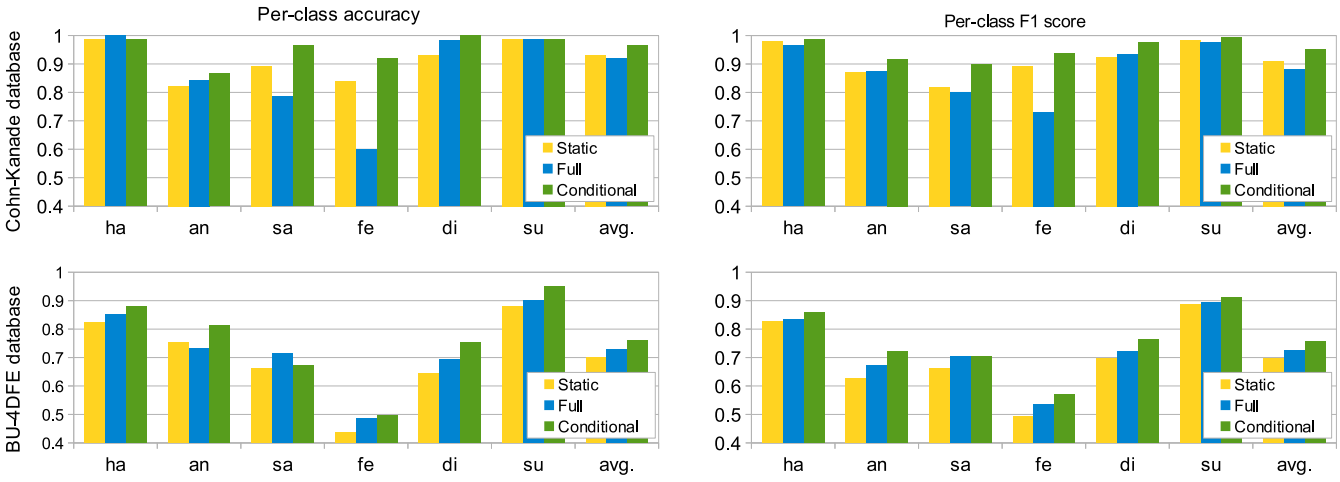


Fig. 8. Per-class recognition accuracy rates and F1-scores on CK+ and BU-4DFE databases

6.3.2 *Cross-database FER on spontaneous data*

In Table 3 we report results for cross-database evaluation (with training on the BU-4DFE database) on the FEED database. In order to provide a fair comparison between our approach and the one presented in [15], we used the same labelling protocol (top 2 accuracy, i.e. we ackowledge correct classification if one of the top 2 proposals outputted by the system matches the ground truth label). The performances of their system are better than those of our static RF model, which can be attributed to the fact that they use a more complex classification and posterior temporal integration flowchart. Nevertheless, our PCRF model provides a substantially higher accuracy ($+3.4\%$), which, again, is likely to be due to the use of spatio-temporal features as well as an efficient conditional integration scheme. Furthermore, modelling spatio-temporal patterns for *every* possible transition (*i.e.* across the videos) allows to gather more training data than using spatio-temporal descriptors [17], [19] learnt on separate videos.

TABLE 3
Comparisons with state-of-the-art approaches on spontaneous data. The first and second best methods are highlighted in red and blue, respectively.

FEED database (Cross-db)	Accuracy
Meguid *et al.* [15]	**53.7**
This work, RF	51.9
This work, PCRF	**57.1**
BP4D database (Cross-db)	Accuracy
Zhang *et al.* [7]	**71.0**
This work, RF	68.6
This work, PCRF	**76.8**

We also performed cross-database evaluation on the BP4D database. Again, for a fair comparison, we used the same protocol as in [7], with training on the BU-4DFE database and using only a subset of the tasks (*i.e.* tasks 1 and 8 corresponding to expression labels *happy* and *disgust* respectively). However, we do not retrain a classifier with a subset of 3 expressions as it is done in [7], but instead use our 7-class static and PCRF models with a forced choice between *happiness* (probability of class *happiness*) and *disgust* (probability sum of classes *anger* and *disgust*). Such setting could theoretically increase the confusion in our conditional model, resulting in a lower accuracy. However, as can be seen in Table 3, using dynamic information within the PCRF framework allows to substantially increase the recognition rate as compared to a static RF framework ($+8.2\%$). We also overcome the results reported in [7] by a significant margin ($+5.8\%$), further showing the capability of our approach to deal with complex spontaneous FER tasks. Also note that in [7], the authors used the so-called *Nebulae* $3D$ polynomial volume features which are by far more computationally expensive than our geometric and integral HOG $2D$ features. All in all, we believe our results show that the PCRF approach provides significant improvements over a traditional static classification pipeline that translates very well to more complicated spontaneous FER scenarios, where a single video may contain samples of several expressions.

6.4 Multi-view experiments

To the best of our knowledge, there is no publicly available benchmark for specifically evaluating dynamic FER methods under head pose variations. We thus propose a new evaluation protocol using the rotated videos generated in Section 5.

6.4.1 *Transition classification*

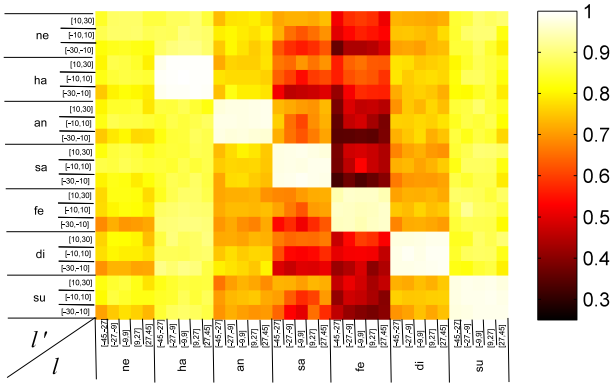


Fig. 9. Accuracies for the classification of transitions going from classes l' to classes l, for every head poses.

First, we evaluate the capacities of our method for transition classification. Specifically, for each pose bin Ω_i and each starting

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

expression label l', we report on Figure 9 the capabilities of a pairwise classifier trained on transitions starting with expression l' and pose Ω_i to distinguish between all expressions l.

One can see that the accuracy is close to 100% on all expressions, in the case where $l' = l$. This is due to the fact that all transitions that stays into an expressive state are characterized by very small deltas of the pairwise features. Transitions towards *fear* are the least successfully recognized, as this class is inherently more subtle and often close to *surprise* (As highlighted in Section 6.3.1). Furthermore, transitions towards classes *anger* and *sadness* are also are also well separated, particularly when the starting expression differ subtly from those expressions (e.g. *neutral*, *anger* and *sadness*), indicating that the pairwise features can successfully capture the subtle differences between those expressions. Finally, transition classification accuracies are generally lower for negative pitches, as it is the case for FER from video.

6.4.2 FER on prototypical data

To perform FER from video, we first estimate the head pose $\omega(x^n)$ for each frame n using the set of aligned feature points. Then, trees from the MVPCRF collections are sampled according to the values $\mathcal{P}_{\Omega_i}(\omega(x^n))$ for each pose bin Ω_i. We compare the average accuracies outputted by RF, PCRF, MVRF and MVPCRF. RF and PCRF were trained on the central (frontal view) bin only. For PCRF and MVPCRF, we set the temporal integration parameters $N = 60$ and $Step = 6$ as it provided satisfying results in the frontal case (Figure 7). As in Section 6.3, a video is considered correctly classified if the dominant expression mode (*i.e.* the maximum probability expression throughout the sequence) corresponds to the ground truth label for that video.

Table 4 displays per-expression accuracies averaged over the 15 pose bins for the three models. For all expressions, MVPCRF outperforms RF and PCRF by a significant margin. MVPCRF also outperforms the static multi-view MVRF on all expressions but *sadness* and *fear*. However, Table 5 reveals that the F1-score is a little higher for MVPCRF on those expressions, indicating that the static MVRF is more biased toward those expression classes. This seems particularly relevant in the positive pitch case, where using spatio-temporal information helps to disambiguate *anger* from *sadness*, which in some case differ only by a very subtle eyebrow frown or lip raiser. Also, *fear* appears as the most subtle expression as already reported in other works [15]. This is due to the fact that subjects often smile during the sequence, thus the videos may be misclassified as *happiness*. For this reason, many other approaches such as the one in [12] use a restricted number of subjects. However, we use the 101 subjects to ensure reproducibility of the results.

The overall classification accuracy is 72.2% against 76.1% for the benchmarks of Section 6.3 on frontal view video. This performance drop comes from a greater variability in face appearance as well as the feature point misalignment for non-frontal poses, as discussed in [44]. Classification rates are also a little lower than the static FER baseline [25], [11] on the BU-3DFE database. However, fully automatic FER from video is a much more difficult setup, as it involves the retrieval of the apex frames and expression classification on those frames. Furthermore, many approaches operate on high-resolution $3D$ data and require expensive projections on a frontal view, thus can not be applied easily to real-time FER from consumer camera.

Figure 10 shows the per-pose bin accuracy rates averaged over the six expressions. On the one's hand, RF performances seems to drop dramatically when we move away from the central bin (from 70.4% to 44.7%). Interestingly, PCRF performs significantly better than RF on every pose bin, which proves that the captured dynamics generalize well on unseen data, as already shown on the cross-database settings. PCRF performance also drops significantly on off-center pose bins. On the other hand, MVPCRF performs significantly better on those bins: accuracy is nearly symetrical for negative and positive yaws, as already reported by [25] for static multi-view FER. Furthermore, as stated in [25], [11] we observe lower classification rates on negative pitches (68.3% as compared to 74.1% on average for positive pitch). Our take is that the mouth area may be the most informative one for FER tasks: as such, the classifiers can struggle to disambiguate certain expressions (e.g. *anger* from *sadness*) when the mouth features become more subtle and difficult to capture.

Figure 11 shows the confusion matrices obtained for RF, PCRF, MVRF and MVPCRF. For each expression, results are divided between the different pose bins to highlight which combinations of view and expressions are well recognized or not. One can see both RF and PCRF struggle to disambiguate the expressions in the case of negative pitches. Using MVRF and MVPCRF allows to compensate for that to an extent, particularly, for instance, in the case of high/low pitches values and expressions *sadness* and *anger*. Still, expressions such as *sadness* and *fear* are better recognized for positive pitches, as they specifically involve subtle mouth movements as well as eyebrow raising. Conversely, *anger* and *disgust* are characterized by eyebrow frowning that is better recognized on negative pitch views. Finally, *happiness* and *surprise* are expressions with the highest overall classification rates. They are typically better recognized on frontal views or for negative pitches, where the corresponding mouth motions are less frequently misclassified as *fear*.

TABLE 4
Per-expression accuracies averaged over all pose bins

Expression	RF (%)	PCRF (%)	MVRF (%)	MVPCRF (%)
Happy	57.8	73.4	83.3	**87.8**
Angry	59.2	73.3	71.9	**80.4**
Sad	56.0	52.2	**70.8**	64.4
Fear	29.6	25.7	**34.8**	33.0
Disgust	48.4	63.9	63.5	**74.3**
Surprise	81.6	88.3	85.3	**92.4**
Average	55.4	62.8	68.3	**72.1**

TABLE 5
Per-expression F1-scores averaged over all pose bins

Expression	RF (%)	PCRF (%)	MVRF (%)	MVPCRF (%)
Happy	62.6	74.4	80.7	**84.2**
Angry	48.6	61.5	62.9	**68.0**
Sad	46.2	48.8	65.4	**66.1**
Fear	34.7	34.7	43.8	**44.8**
Disgust	56.3	66.2	67.9	**73.1**
Surprise	71.4	77.6	83.6	**87.3**
Average	53.3	60.6	67.4	**70.6**

6.4.3 Cross-database FER "in the wild"

In order to evaluate the capabilities of MVPCRF for FER in unconstrained scenarios, we report in Table 6 the accuracies obtained for cross-database evaluation on the validation set of AFEW, with training on the BU-4DFE database. We report top 1 accuracy for comparison with the challenge baseline [45], as well

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAFFC.2017.2708106, IEEE Transactions on Affective Computing

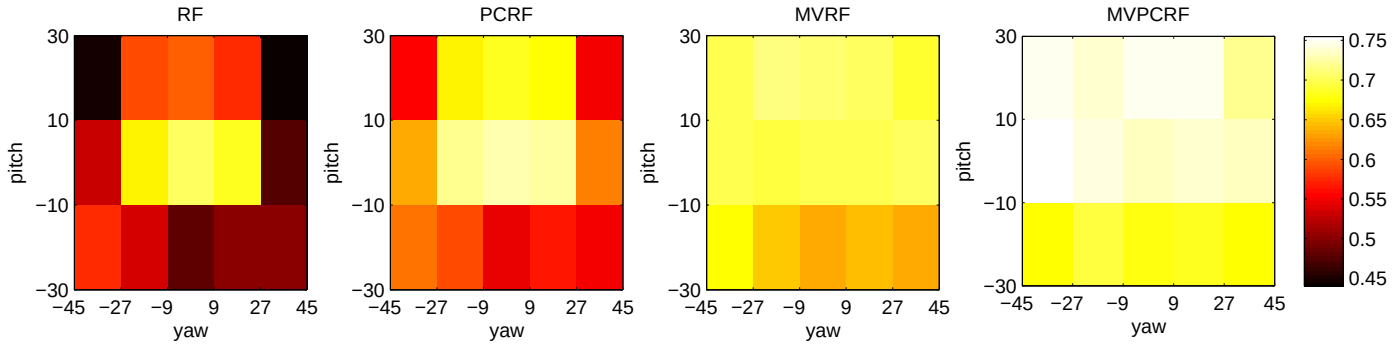


Fig. 10. Per-pose bin accuracy rates averaged over all expressions

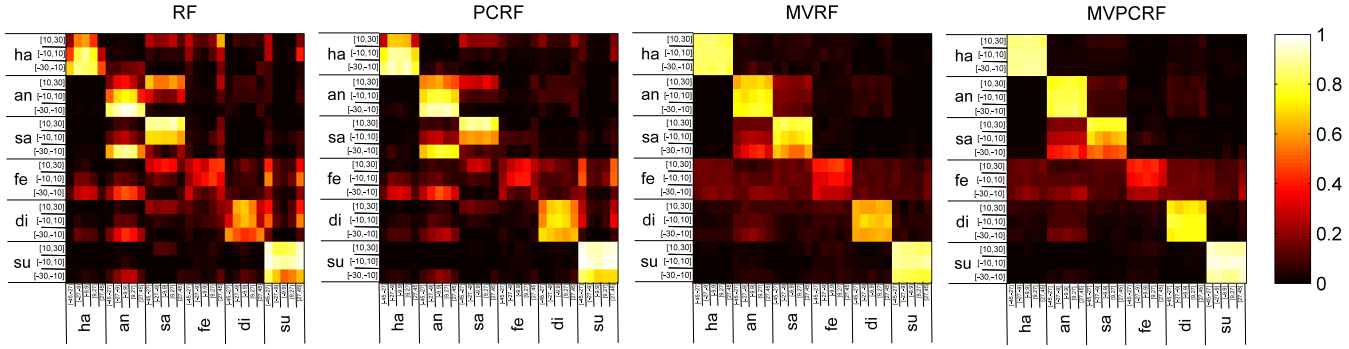


Fig. 11. Per-expression, per-pose bin classification accuracy rates

as top 2 accuracy for comparison with Meguid *et al.* [15], who also perform a cross-database evaluation. Even though the accuracies are very low, particularly in the case of top 1 evaluation, the results are above those of a baseline (LBP-TOP) system trained on AFEW. The top 2 accuracy is also better than the one reported in [15]. Interestingly, due to the difficulty of the task, more than 20% of the total number of video frames were missed by the feature point tracker (and many more were badly aligned). This reflects the advantage of using pairwise classification, which allows to process discontinuous sequences and flexibly handle failure cases, as discussed in Section 6.6.

TABLE 6
Evaluation on the AFEW database (%)

Expression	top 1	Baseline [45]	top 2	Meguid *et al.* [15]
Neutral	16.7	**63.6**	**50.0**	28.6
Happy	**38.7**	25.8	**50.0**	40.0
Angry	18.7	**44.1**	30.5	**56.3**
Sad	**30.7**	7.8	**51.6**	29.3
Fear	**11.1**	5.6	51.9	**56.9**
Disgust	**32.0**	0.0	54.0	65.1
Surprise	**27.5**	5.8	**41.2**	35.6
Average	**25.1**	22.2	**47.1**	44.0

6.5 Complexity analysis

An advantage of using conditional models is that with equivalent parallelization they are faster to train than a full model learnt on the whole dataset. According to [46] the average complexity of training a RF classifier with M trees is $\mathcal{O}(MKN\log^2 N)$, with K being the number of features to examine for each node and N the size of (2/3 of) the dataset. Thus if the dataset is equally divided into P bins of size $\tilde{N}$ upon which conditional forests are trained (and such that $N = P\tilde{N}$), the average complexity of learning a conditional model now becomes $\mathcal{O}(MKN\log^2 \tilde{N})$. Same considerations can be made concerning the evaluation, as trees from the full model are bound to be deeper than those from the conditional models. Table 7 shows an example of profiling a MVPCRF on one video frame with an averaging over 60 frames and a step of 6 frames. We experiment with various total numbers of trees M to show that the proposed framework can perform real-time FER.

TABLE 7
Profiling of total processing time for one frame (in ms)

Step	Time (ms)
Facial alignment	10.0
Integral HOG channels computation	2.0
MVPCRF evaluation (M = 500)	2.6
MVPCRF evaluation (M = 1000)	4.8
MVPCRF evaluation (M = 2000)	7.8
MVPCRF evaluation (M = 6000)	19.0

This benchmark was conducted on a *I7-4770* CPU within a C++/OpenCV environment, without any code parallelization. As such, the algorithm already runs in real-time. Furthermore, evaluations of pairwise classification or tree subsets can be parallelized to fit real-time processing requirements on low-power engines such as mobile phones. In addition, the facial alignment step can be performed at more than 300 fps on a smartphone with similar performances using the algorithms from [28].

6.6 Discussion

The novel approach introduced in this work, which consists in combining the output predictions of pairwise classifiers, offers some advantages over traditional methods, and also suffers from a number of limitations.

First, even though testing can be performed in a fully automatic fashion, peak frame annotations are mandatory for training. In our work, we used images around manually highlighted peak frames to train our system. Hence, the proposed transition modelling approach is robust to noise in the peak frame selection process to a certain extent. As is, requiring more precise labelling for training is a recurrent drawback of frame-based classifiers as compared to sequence-level ones (e.g. HMMs, CRFs) and thus could not be solved easily. However, it would be interesting to study the use of weakly-labelled conditional transition classifiers to alleviate this problem (for example, cluster the images based on the automatically retrieved mouth opening). Also note that using only a subset of the videos for training allows to limit the memory usage, which is particularly relevant when training multi-view classifiers upon large databases. Moreover, an advantage of integrating spatio-temporal information under the form of transition modelling is that it does not require continuity of the sequence, as showed in Section 6.4.3. Hence, it has no problem handling failure from the detection or feature point alignment pipelines, as opposed to other spatio-temporal descriptors [17], [19] and graphical models such as HMMs or LOMO [22].

Secondly, in order to build PCRF and MVPCRF we need examples for each combination of facial expression and head pose. This can be a hindrance when training on highly unbalanced datasets (as in CK+ with *contempt* expression class). In this paper, we circumvented this problem by using high-resolution $3D$ models to generate training examples. Should this data not be available, the number of subdivisions could in theory be limited. In such a case, for example, the transition classifiers could be conditioned on a restricted set of coarser clusters involving closely related expressions (*fear* and *surprise* together, *neutral*, *anger* and *sadness*) or merged adjacent head poses. Moreover, even though using conditional models results in more memory usage, the system benefits from lower runtimes for both training and testing, as showed in Section 6.5.

Conclusion

In this paper, we presented an adaptation of the Random Forest framework for automatic dynamic pose-robust facial expression recognition from videos. We also introduced a novel way of integrating spatio-temporal informations by considering pairwise RF classifiers. This formulation allows the efficient integration of high-dimensional, low-level spatio-temporal information through averaging over time pairwise trees. These trees are conditioned on predictions outputted for the previous frames to help reducing the variability of the ongoing transition patterns. In addition, we proposed an extension of the PCRF framework to efficiently handle head pose variation in an expression recognition system. We showed that our models can be trained and evaluated efficiently given appropriate data, and lead to a significant increase of performances compared to a static RF. We also introduced a new multi-view video corpus generated using the BU-4DFE database to assess the pose-robustness of the proposed system. The `Matlab` code used to render the images that we used for training and testing the classifiers will be made publicly available. Finally, we showed that our method works on real-time without specific optimization schemes, and could be run on low-power architectures such as mobile phones by using appropriate paralellization scheme.

As such, future works will consist in addressing occlusions to better adapt MVPCRF for FER in the wild [39]. As $3D$ models have already been very useful for generating data for a variety of problems, we believe that robustness to self-occlusions could also benefit from such data. Furthermore, we would like to investigate applications of transition modelling for other video classification/regression problems such as Facial Action Unit intensity prediction or body and hand gesture recognition.

Acknowledgments

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

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

[2] E. Sariyanidi, H. Gunes, and A. Cavallaro, "Automatic analysis of facial affect: A survey of registration, representation and recognition," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1113 – 1133, 2014.

[3] P. Ekman and W. Friesen, "Constants across cultures in the face and emotion." *Journal of personality and social psychology*, vol. 17, no. 2, p. 124, 1971.

[4] 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 *International Conference on Computer Vision and Pattern Recognition Workshops*, 2010, pp. 94–101.

[5] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato, "A 3D facial expression database for facial behavior research," in *International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 211–216.

[6] L. Yin, X. Chen, Y. Sun, T. Worm, and M. Reale, "A high-resolution 3D dynamic facial expression database," in *International Conference on Automatic Face and Gesture Recognition*, 2008, pp. 1–6.

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

[8] M. K. Greenwald, E. W. Cook, and P. J. Lang, "Affective judgment and psychophysiological response: Dimensional covariation in the evaluation of pictorial stimuli," *Journal of psychophysiology*, vol. 3, no. 1, pp. 51–64, 1989.

[9] P. Ekman and W. V. Friesen, "Facial action coding system," *Consulting Psychologists Press, Stanford University*, 1977.

[10] J. F. Cohn, "Foundations of human computing: facial expression and emotion," in *International Conference on Multimodal Interfaces*, 2006, pp. 233–238.

[11] R.-L. Vieriu, S. Tulyakov, S. Semeniuta, E. Sangineto, and N. Sebe, "Facial expression recognition under a wide range of head poses," in *International Conference on Automatic Face and Gesture Recognition*, 2015.

[12] B. Ben Amor, H. Drira, S. Berretti, M. Daoudi, and A. Srivastava, "4-D facial expression recognition by learning geometric deformations," *Transactions on Cybernetics*, vol. 44, no. 12, pp. 2443–2457, 2014.

[13] R. A. Khan, A. Meyer, H. Konik, and S. Bouakaz, "Human vision inspired framework for facial expressions recognition," in *International Conference on Image Processing*, 2012, pp. 2593–2596.

[14] S. Happy and A. Routray, "Automatic facial expression recognition using features of salient facial patches," *Transactions on Affective Computing*, pp. 1–13, 2014.

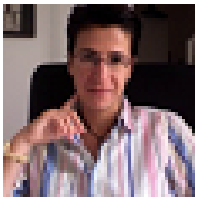
[15] M. Abd El Meguid and M. Levine, "Fully automated recognition of spontaneous facial expressions in videos using random forest classifiers," *Transactions on Affective Computing*, vol. 5, pp. 151–154, 2014.

[16] M. Mohammadi, E. Fatemizadeh, and M. Mahoor, "Non-negative sparse decomposition based on constrained smoothed l0 norm," *Signal Processing*, vol. 100, pp. 42–50, 2014.

[17] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, 2007.
[18] M. Hayat, M. Bennamoun, and A. El-Sallam, "Evaluation of spatiotemporal detectors and descriptors for facial expression recognition," in *International Conference on Human System Interactions*, 2012, pp. 43–47.
[19] A. Klaser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3D-gradients," in *British Machine Vision Conference*, 2008, pp. 995–1004.
[20] S. Shojaeilangari, W.-Y. Yau, J. Li, and E.-K. Teoh, "Multi-scale analysis of local phase and local orientation for dynamic facial expression recognition," *Journal of Multimedia Theory and Application*, vol. 1, pp. 1–10, 2014.
[21] Z. Wang, S. Wang, and Q. Ji, "Capturing complex spatio-temporal relations among facial muscles for facial expression recognition," in *International Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3422–3429.
[22] K. Sikka, G. Sharma, and M. Bartlett, "Lomo: Latent ordinal model for facial analysis in videos," *International Conference on Conference on Computer Vision and Pattern Recognition*, 2016.
[23] A. Dapogny, K. Bailly, and S. Dubuisson, "Pairwise conditional random forests for facial expression recognition," *International Conference on Computer Vision*, pp. 1–9, 2015.
[24] W. Zheng, H. Tang, Z. Lin, and T. S. Huang, "Emotion recognition from arbitrary view facial images," in *European Conference on Computer Vision*, 2010, pp. 490–503.
[25] U. Tariq, J. Yang, and T. S. Huang, "Multi-view facial expression recognition analysis with generic sparse coding feature," in *European Conference on Computer Vision Workshops*, 2012, pp. 578–588.
[26] S. Eleftheriadis, O. Rudovic, and M. Pantic, "Discriminative shared gaussian processes for multiview and view-invariant facial expression recognition," *Transactions on Image Processing*, vol. 24, no. 1, pp. 189–204, 2015.
[27] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *International Conference on Computer Vision and Pattern Recognition*, 2013, pp. 532–539.
[28] S. Ren, X. Cao, Y. Wei, and J. Sun, "Face alignment at 3000 fps via regressing local binary features," in *International Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1685–1692.
[29] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, "Face alignment across large poses: A 3d solution," *International Conference on Computer Vision and Pattern Recognition*, 2016.
[30] M. Dantone, J. Gall, G. Fanelli, and L. Van Gool, "Real-time facial feature detection using conditional regression forests," in *International Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2578–2585.
[31] H. Yang and I. Patras, "Privileged information-based conditional regression forest for facial feature detection," in *International Conference on Automatic Face and Gesture Recognition*, 2013, pp. 1–6.
[32] M. Sun, P. Kohli, and J. Shotton, "Conditional regression forests for human pose estimation," in *International Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3394–3401.
[33] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
[34] J. R. Quinlan, *C4. 5: programs for machine learning*, 2014.
[35] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *British Machine Vision Conference*, 2009.
[36] D. F. Dementhon and L. S. Davis, "Model-based object pose in 25 lines of code," *International Journal of Computer Vision*, vol. 15, no. 1-2, pp. 123–141, 1995.
[37] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 511–518.
[38] F. Wallhoff, "Database with facial expressions and emotions from technical university of munich (feedtum)," http://cotesys.mmk.e-technik.tu-muenchen.de/waf/fgnet/feedtum.html, 2006.
[39] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon, "Acted facial expressions in the wild database," *Australian National University, Canberra, Australia, Technical Report TR-CS-11-02*, 2011.
[40] T. Bylander, "Estimating generalization error on two-class datasets using out-of-bag estimates," *Machine Learning*, vol. 48, no. 1-3, pp. 287–297, 2002.
[41] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *Winter Conference on Applications of Computer Vision*, 2016, pp. 1–10.
[42] P. Liu, S. Han, Z. Meng, and Y. Tong, "Facial expression recognition via a boosted deep belief network," in *International Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1805–1812.
[43] Y. Sun and L. Yin, "Facial expression recognition based on 3D dynamic range model sequences," in *European Conference on Computer Vision*, 2008, pp. 58–71.
[44] L. Jeni, D. Takacs, A. Lorincz *et al.*, "High quality facial expression recognition in video streams using shape related information only," in *International Conference on Computer Vision Workshops*, 2011, pp. 2168–2174.
[45] A. Dhall, R. Goecke, J. Joshi, M. Wagner, and T. Gedeon, "Emotion recognition in the wild challenge 2013," in *International Conference on Multimodal Interaction*, 2013, pp. 509–516.
[46] G. Louppe, "Understanding random forests: From theory to practice," Ph.D. dissertation, University of Liège, 2014.

Arnaud Dapogny is a PhD student at Institute for Intelligent Systems and Robotics (CNRS UMR 7222), University Pierre and Marie Curie-Paris 6. He also obtained the Engineering degree from the SUPELEC Engineering School in 2011 and the Masters degree from University of Paris 6 in 2013 with high honors. His works concern face and gesture analysis as well as automatic real-time facial expression recognition.

Kévin Bailly is an Associate Professor at UPMC Sorbonne Universities and a researcher at the Institute for Intelligent Systems and Robotics (CNRS UMR 7222). He received the Ph.D. degree in Computer Science from UPMC in 2010 and was a research fellow at telecom ParisTech LTCI lab in 2010/2011. His research interests are machine learning and computer vision applied to face processing and behaviour analysis. He won several challenges on automatic facial expression analysis (FERA'11, FERA'15 and AVEC'12).



Séverine Dubuisson was born in 1975. She received the Ph.D. degree in system control from the Compigne University of Technology, France, in 2001. From 2002 to 2013, she has been an Associate Professor with the Laboratory of Computer Sciences (LIP6), UPMC Sorbonne Universits, France. Se is now associate professor in Institut for Intelligent Systems and Robotics (ISIR), UPMC Sorbonne Universits, France. Her research interests include computer vision, visual tracking, probabilistic models for video sequence analysis and human interaction.