Contents lists available at ScienceDirect

Journal of Visual Communication and Image Representation

journal homepage: www.elsevier.com/locate/jvci

Facial micro-expression recognition based on accordion spatio-temporal representation and random forests☆

Radhouane Guermazi [a,*], Taoufik Ben Abdallah [b], Mohamed Hammami [c]

[a] Saudi Electronic University, Riyadh, Saudi Arabia
[b] MIR@CL Laboratory, Faculty of Economics and Management of Sfax, Tunisia
[c] MIR@CL Laboratory, Faculty of Sciences of Sfax, Tunisia

ARTICLE INFO

Keywords:
Facial micro-expressions
Pyramid levels' combination
Accordion representation
Low-dimensional feature space
Random Forests
Proximity measure

ABSTRACT

Micro-expressions are very brief involuntary facial expressions which appear on the face of humans when they unconsciously conceal an emotion. Creating a solution allowing an automatic recognition of the facial micro-expressions from video sequences has garnered increasing attention from experts across such different disciplines as computer science, security, and psychology. This paper offered a solution to facial micro-expressions recognition, based on accordion spatio-temporal representation and Random Forests. The proposed feature space, called "Uniform Local Binary Patterns on an Accordion 2D representation of sub-regions presented by a Pyramid of levels (LBPAccP^{u2})", exploits the effectiveness of uniform LBP patterns applied on an accordion representation of sub-regions at different sizes. Random Forests were used to select the most discriminating features and reduce the classification ambiguity of similar micro-expressions through a new proximity measure. The main objective of our paper was to demonstrate that the use of few features could be more efficient to produce a strong micro-expression recognition classifier that outperforms the approaches that rely on high dimensional features space. The experimental results across six micro-expression datasets show the effectiveness of the proposed solution with an accuracy rate that can reach 81.38% on CasmeII dataset. Compared to some famous competitive state-of-the-art approaches, the proposed solution proved its performance thanks to its accuracy rate as well as the number of features it uses.

1. Introduction

Communication involves the words we use to exchange information, news, ideas and feelings. However, the interpersonal communication is much more than the explicit meaning of words. It also includes hidden messages, expressed through facial expressions, gestures, and tone of voice. These non-verbal signals represent the non-verbal communication. Unlike most of the non-verbal communication forms that can vary from one culture to another, facial expressions are universal. No matter where a person comes from or what language he speaks, his smile may indicate happiness or approval and his frown may indicate unhappiness or disapproval. Researcher Paul Ekman defines six universal emotions that are: *disgust, sadness, happiness, fear, anger, and surprise* [1]. These facial expressions can roughly be divided into two categories: facial macro-expressions and facial micro-expressions.

Facial macro expressions typically last for more than half a second. We see them in our daily interactions with people all of the time. In contrast, facial micro-expressions reveal the true feeling that people are trying to hide or inhibit. They are too brief. They last less than 1/25th of a second and cannot be easily detected in real conversations.

While facial micro-expression recognition is a challenging task [2], it is widely used in many fields such as police services and lies detection [3], business office [4,5], clinical diagnosis [4,6], teaching assistance [7] and teaching engagement assessment [8].

Over the past few decades, many researchers have made huge efforts to help computer better understand facial micro-expressions and emotional communication among humans. The literature on this field shows a variety of methods including face detection and tracking, preprocessing, facial feature detection and selection, and classification [2]. These methods can be decomposed into (i) appearance-based methods, (ii) motion-based methods, and (iii) deep learning-based methods.

Through the literature, we can notice that more attention has been

☆ This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author.
E-mail address: r.guermazi@seu.edu.sa (R. Guermazi).

https://doi.org/10.1016/j.jvcir.2021.103183
Received 31 August 2020; Received in revised form 15 March 2021; Accepted 5 June 2021
Available online 23 June 2021

paid to appearance-based methods. These methods describe texture information such as wrinkles, furrows, and intensity dynamics of expressions. Several representations were used to characterize the facial micro-expressions such as the Gabor filters [9], the 2D Gabor filter and Sparse Representation (2DGSR) [10], the Discriminant Tensor Subspace Analysis (DTSA) [11], and the Local Binary Patterns on Three Orthogonal Planes (LBP-TOP) [12–16]. Several studies proposed variants of LBP-TOP to amplify the facial micro-expression information in a very brief time [17,18,15,19,20]. For example, Guo et al., [17] proposed the Centralized Binary Patterns on Three Orthogonal Planes (CBP-TOP). They considered only the pixels with the highest weight among their neighboring points. Huang et al., [21] proposed the Spatio-Temporal Local Binary Patterns with Integral Projection (STLBP-IP). Indeed, they used integral projection to preserve the shape features of facial images and therefore increase the discrimination for the facial micro-expressions. To go further, Huang et al., [18] proposed the Spatio-Temporal Completed Local Quantized Patterns (STCLQP). They have exploited orientation, sign and magnitude components to provide appropriate pattern codes instead of uniform patterns. The use of these pattern codes can achieve better performance, but depends on the training dataset used for building classifiers. Wang et al., [15] used LBP-TOP to recognize the facial micro-expressions after preprocessing the dataset with the Eulerian Video Magnification (EVM), which consists of enhancing the low intensity of the facial micro-expressions. Liong et al., [20] selected some facial regions statically based on AUs frequency occurring (ROI-selective). Zong et al., [19] argued that the conventional spatial division method cannot ensure such appropriate sub-regions because its grid size is fixed. As a matter of fact, using large sub-regions may produce noisy information and interfere with the performance of spatiotemporal features, while small sub-regions lead to the loss of useful information. To solve this problem, they proposed a hierarchical division scheme in which they divided face into sub-regions with different sizes. They applied STLBP-IP, as in [21], to detect appearance features. The proposed feature space can be more competitive and powerful in dealing with facial micro-expression recognition.

To overcome the problem of limited labeled facial micro-expression samples, Jia et al., [22] and Ben et al., [23] exploited the macro-expression samples for improving the recognition of three facial micro-expressions: *"disgust"*, *"happiness"* and *"surprise"*. Jia et al., [22] used LBP (reps. LBP-TOP) to detect the facial macro-expression (resp. micro-expression) features and employed the Singular Value Decomposition algorithm (SVD) to achieve a macro-to-micro transformation model. Ben et al., [23] proposed the Hot Wheel Patterns (HWP) and HWP-TOP for facial macro/ micro-expression representation, and employed the Coupled Metric Learning algorithm (CML) to model the shared features between facial micro-expression samples and facial macro-information. Compared to the other state-of-the art approaches that are based only on micro-information, using macro-expression information to recognize the facial micro-expressions does not record a noticeable improvement in effectiveness.

Several publications have appeared in recent years documenting the motion-based methods which measure non-rigid movement and orientations of the facial component [24–29]. In this category of methods, the Histogram of Oriented Optical Flow (HOOF) and its variants have been widely applied for facial micro-expression recognition via the computation of facial movement changes between consecutive frames using the brightness conservation principle [30]. The Histogram of Oriented Gradient (HOG) and its variants have also been applied in order to express motion changes locally and between adjacent frames. Li et al., [28] proposed the Histogram of Image Gradient Orientation on Three Orthogonal Planes (HIGO-TOP) to describe motion changes of the facial micro-expressions. In fact, they employed a magnification on frames to improve the feature effectiveness.

In the last few years, there has been a growing interest in using the deep learning-based methods to recognize facial micro-expressions. The proposed methods are performed on different deep neural network architectures such as the Convolutional Neutral Network (CNN) and the Recurrent Neural Network (RNN). They tried to simultaneously achieve a high-level feature detection and pattern recognition [31–33].

One of the first uses of the deep learning-based methods in facial micro-expression recognition was carried out by Kim et al., [31]. They applied CNN on some frames of a video sequence at different expression-states (i.e. onset, apex and offset) to detect a high-level-spatial feature space. They also used the Long Short-Term Memory (LSTM), which derived from RNN, to detect time scale dependent features from the video sequences. Peng et al., [32] proposed the Dual Temporal Scale Convolutional Neural Network (DTSCNN) for facial micro-expression recognition. They took optical-flow sequences in different temporal scales as the input of DTSCNN in order to calculate the high-level spatio-temporal feature space. Wang et al., [33] applied the Transferring Long-term Convolutional Neural Network (TLCNN) to extract features from each frame of video sequences. Moreover, Reddy et al., [34] proposed a 3D-CNN architecture from video sequence based on 3D kernel for convolution and feature detection. The softmax layer is used to generate the class scores for the classes of the dataset being used.

Recently, Takalkar et al., [35] proposed a "hybrid" approach based on handcrafted and deep features to recognize micro-expressions. The handcraft features, based on LBP-TOP, represent the spatio-temporal movements of the face, while the deep features are extracted using CNN.

Under the deep learning-based methods, the recognition process is carried out in a "black box". Despite the emergence of the deep learning-based methods and the effective classifiers in facial macro-expression recognition [36,37], their efficiency is still limited on facial micro-expression recognition due to the small size of the facial micro-expression datasets. Indeed, a large amount of data is required to obtain discriminating high-level features and avoid the problem of overfitting. In this context, Zhi et al., [38] and Wang et al., [39] used the transfer learning to deal with the problem of insufficient samples in the facial micro expression dataset. They pre-trained their model on a facial macro-expression dataset. Then, they transferred and fine-tuned their model to recognize facial micro-expression. Compared to deep learning-based methods, the transfer learning-based methods do not improve too much the accuracy of micro-expressions recognition which always remains non-competitive. The same issue can be observed with the motion-based methods. Indeed, their major drawback is their performance decay due to noise like brightness, non-aligned face, and the extremely short duration and subtle movement of micro-expressions [2].

One of the big issues with the appearance-based methods is the high dimension of the feature space. For example, Huang et al., [18] used more than 23000 features. Likewise, Wang et al., [15] created their solution using more than 4425 features. To deal with this issue, Wang et al., [40] designed 16 Regions of Interests (ROIs) based on Facial Action Coding System (FACS) [41] for feature detection. Also, Liu et al., [24] and zong et al.,[19] used some ROIs for facial micro-expression recognition. They proved that the appearance changes caused by the facial micro-expressions only emerge in a few local and small facial sub-regions [41]. Experiments showed that using the features detected from ROIs improves the facial micro-expression effectiveness. Moreover, Abdallah et al., [42] proposed a low-dimensional feature space based on the Pyramid of uniform Local Binary Patterns (PLBP^{u2}) [43]. The proposed space seems to be more efficient in recognizing the facial macro-expressions than facial micro-expressions. It is worth to mention that as the deep-learning methods and the motion-based methods, the efficiency of the appearance-based method in the recognition of micro-expressions remains limited even when their feature space is detected from ROIs. We speculate that this might be due to the ambiguity of classification of similar facial micro-expressions which can be considered as one of the tough challenges for all researchers in this domain.

To deal with the different issues of the related works, we proposed in this paper an innovative approach of facial micro-expression recognition-based on a proposed appearance feature space, that we call "Uniform Local Binary Patterns on an Accordion 2D representation of sub-regions presented by a Pyramid of levels (LBPAccP^{u2})" and a proposed classification algorithm, based on Random Forests (RF) and a proximity measure, referred to as *RF_prox*. This research work focused not only on the efficiency of the proposed approach in facial micro-expression recognition but also on the dimension of the used feature space. The originality of the proposed solution lies in the fact that the use of few features and an adapted classifier can be more efficient to produce a strong facial micro-expression recognition classifier that outperforms the performance of the famous literature approaches.

The proposed feature space exploits the effectiveness of the uniform LBP patterns from an accordion representation of sub-regions at different sizes. This representation transforms a video sequence into a 2D image that keeps both spatial and temporal information sets. Contrary to the majority of the literature methods that use a feature selection method to reduce the dimension of their feature space, our approach reduces the number of features by using an adequate video representation method and a feature selection method. In fact, compared to the 3D frame-representation, the accordion representation decreases the number of features used to present a video sequence. The RF is also used to select the most discriminating features from the proposed space. The number of selected features is remarkably small. Experiments using the proposed low-dimensional feature space prove its performance to categorize facial micro-expressions better than using the whole space.

The proposed classification algorithm is designed to reduce the ambiguity of classification of similar facial micro-expressions like *"disgust"* and *"repression"* by using a proximity measure calculated from the RF. As a matter of fact, RF is used to overcome the issue of the small size of the facial micro-expression datasets and the risk of overfitting in building classifiers.

The remainder of this paper is organized as follows: Section 2 details the proposed approach. Section 3 presents the experimental framework. Section 4 discusses the experimental results. Section 5 presents a comparative study. Section 6 ends up with some concluding remarks and research perspectives.

2. Proposed approach

The aim of the proposed approach is to recognize a frontal pose-invariant facial micro-expression video using a low-dimensional appearance feature space. A flowchart of the proposed approach is illustrated in Fig. 1. It consists of four main parts: (i) preprocessing the input video sequence by detecting and tracking the face; (ii) representing the whole of the video sequence using the proposed feature space, called "Uniform Local Binary Patterns on an Accordion 2D representation of sub-regions presented by a Pyramid of levels (LBPAccP^{u2})"; (iii) selecting the most discriminating features using an embedded feature selection method based on Random forests (RF) [44]; and (iv) building the micro-expression classifier based on the proposed *RF_prox* algorithm.

In the face dection and tracking step, we apply the Viola and Jones's algorithm because it can be considered as the most used algorithm for face detection in AFER. This algorithm incorporates the Adaboost classification technique to select the most discriminating features as well as to train the classifiers. the first frame is used for automatic face detection, while a face tracking is applied for the rest of the frames.

The following sub-sections give an overview of the algorithms proposed for feature detection, selection, and classification.

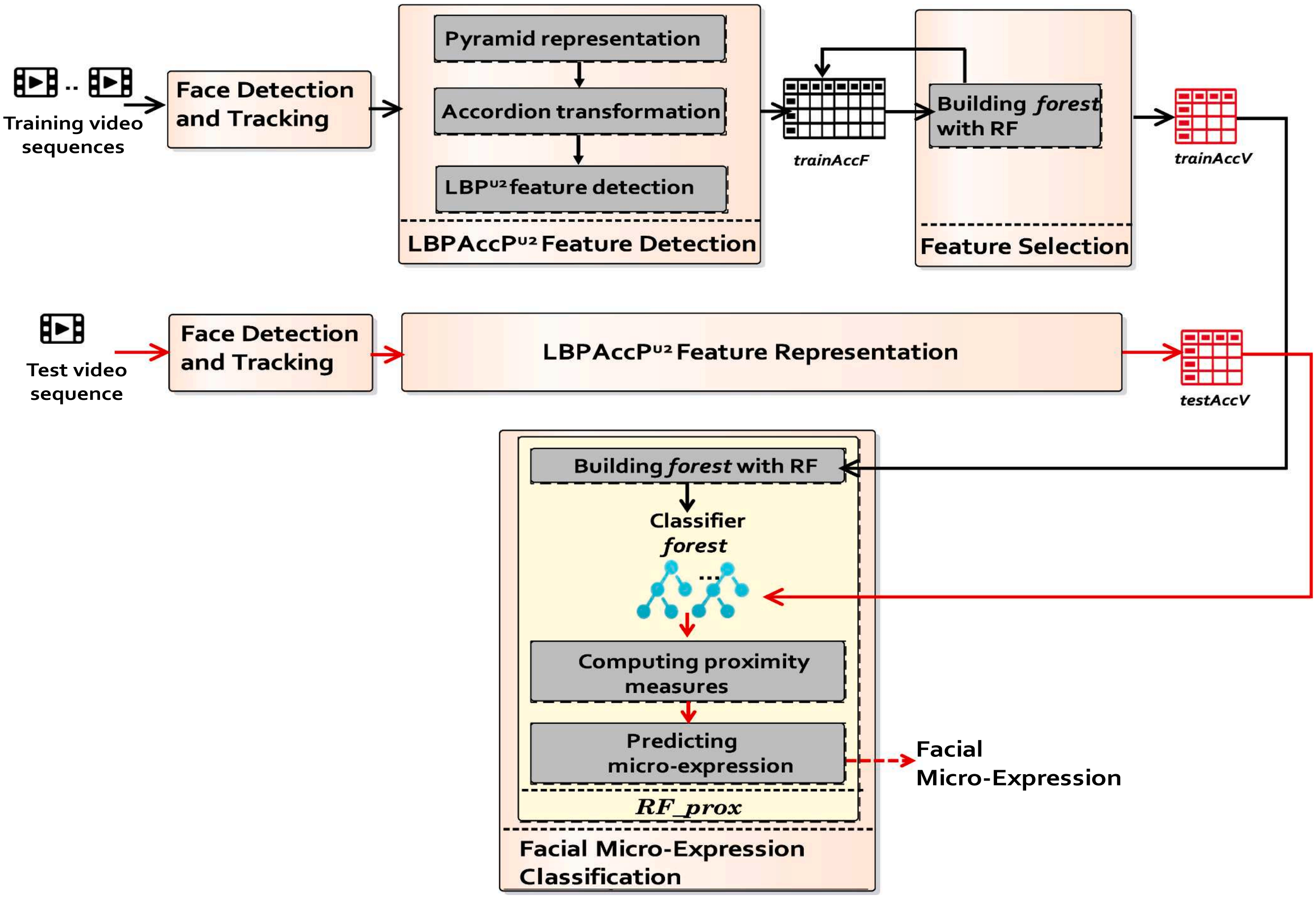


Fig. 1. An overview of the proposed facial micro-expression recognition approach.

2.1. LBPAccP^{u2} feature detection

In this subsection, we introduce our proposed feature space, called LBPAccP^{u2}. It is based on the uniform Local Binary Patterns (LBP^{u2}) [45] applied on a 2D spatio-temporal representation of the sub-regions at different sizes that are extracted according to the pyramid levels. The representation of a video sequence as a 2D image is generated by the accordion technique, proposed for video compression by Ouni et al., [46]. Our objective is to project temporal redundancy of all the video frames and combine them with spatial redundancy into one high spatial correlation representation. Unlike other representations in the literature that provide high-dimensional feature spaces [13,18], the use of the accordion technique allows decreasing the number of features necessary to present a video sequence. Furthermore, the pyramid levels are used to give a different description of the facial appearance changes. The use of pyramid representation can assign more importance to some sub-regions rather than others. Also, the use of LBP^{u2} makes the feature vector much smaller and reduces the number of codes inflicted by a high-frequency noise as well.

Algorithm 1. LBPAccP^{u2} feature detection

```
Input : video: 3D pixel matrix of a video sequence.
        pLev: vector containing the indices of the pyramid
levels.
begin
    $F \leftarrow [\ ]$; /*F: LBPAccP$^{u2}$ feature vector*/
    /*Representing video as a set of cuboids according to pLev*/
    $data\{data_1, \ldots, data_{NR}\} \leftarrow$
      $representPyramid(video, pLev)$;
    for $r$=1 to $NR$ do
        /*Representing each cuboid in accordion*/
        $accData_r \leftarrow transformAccordion(data_r)$;
        /*Computing LBP$^{u2}$ histogram*/
        $hist_r \leftarrow calculateLBPfeatures(accData_r)$;
        /*Concatenating horizontally LBP$^{u2}$ histograms*/
        $F \leftarrow [F, hist_r]$;
```

As Shown in Algorithm 1, the process of LBPAccP^{u2} is performed in four main steps:

- The first step aims at decomposing each video sequence into sub-regions (i.e. cuboids) at different sizes according to the pyramid levels. This step is carried out by invoking the function *representPyramid*(*video*, *pLev*), where *video* denotes the 3D pixel matrix of a video sequence, and *pLev* is the μ-dimensional vector containing the indices of levels that will be considered in the pyramid representation. For instance, $pLev = [2, 3]$ presents two levels ($\mu = 2$) in the pyramid: $pLev(1)$ refers to level 2 and the second level $pLev(2)$ corresponds to level 3. The output of the function *representPyramid*(*video*, *pLev*), referred as *data*, represents the set of cuboids. It is worth to mention that the number and the size of cuboids depend on *pLev*. Given $H \times L \times NF$ the size of *video*, the level l contains $NR = 4^l$ cuboids. The size of each cuboid is $\frac{L}{2^l} \times \frac{H}{2^l} \times NF$ pixels. Fig. 2 presents an example of a pyramidal representation of three levels ($pLev = [1,2,3]$). As shown in this figure, the cuboid 1 in the level 1 is represented by the cuboids 1, 2, 5 and 6, in the level 2 and the cuboids 1,2,3,4,9,10,11,12,17,18,19,20,25,26,27, and 28 in the level 3. Let's suppose that the selected features correspond to the cuboid 1 from level 1 and the cuboid 5 from level 2 and the cuboid 13 from level 3. In this case, the cuboid 13 will be considered 3 times (in all the levels) and the cuboid 5 will be considered 2 times (in level 2 and in level 1).
- The second step focuses on transforming of each cuboid, referred to as *3DMatrix*, into a 2D image using the accordion representation [46]. This step is carried out by calling the function *transformAccordion*(*3DMatrix*). It returns a 2D-pixel matrix, referred to as *2Dmatrix*, which is represented by H rows and $L \times NF$ columns (Fig. 3). Formally, we extract the first column of each frame of the video and we put them in the *NF* first columns of *2DMatrix*, in the normal order, going from the first frame to the last one. The second column of each frame of the video is represented in the *NF* second columns of the *2DMatrix*, in the reverse order, going from the last frame to the first one. The third column is treated like the first one, the fourth like the second one, and so on.
- The third step calculates the LBP^{u2} [47] from each *2DMatrix* of each sub-region based on the *calculateLBPfeatures*(*2DMatrix*) function. In our work, based on the study proposed by [47], the number of P neighboring pixels equals 8 and the radius R equals 1. Likewise, we have not considered the extremities' pixels whose coordinates $x \notin [R + 1 \cdots H - R]$and $y \notin [R + 1 \cdots L - R]$, where H and L are the height and the length of the frame of a video sequence respectively.
- The fourth step consists of concatenating horizontally each LBP^{u2} histogram of the *NR* sub-regions, determined according to the pyramid levels, to obtain the LBPAccP^{u2} feature space of dimension *nbF* equals 59 × *NR*.

2.2. Feature selection based on RF

In our work, we have applied RF [44] in order to select the most discriminating features from LBPAccP^{u2} feature space as illustrated in Algorithm 2.

Algorithm 2. Feature selection with RF

(continued on next page)

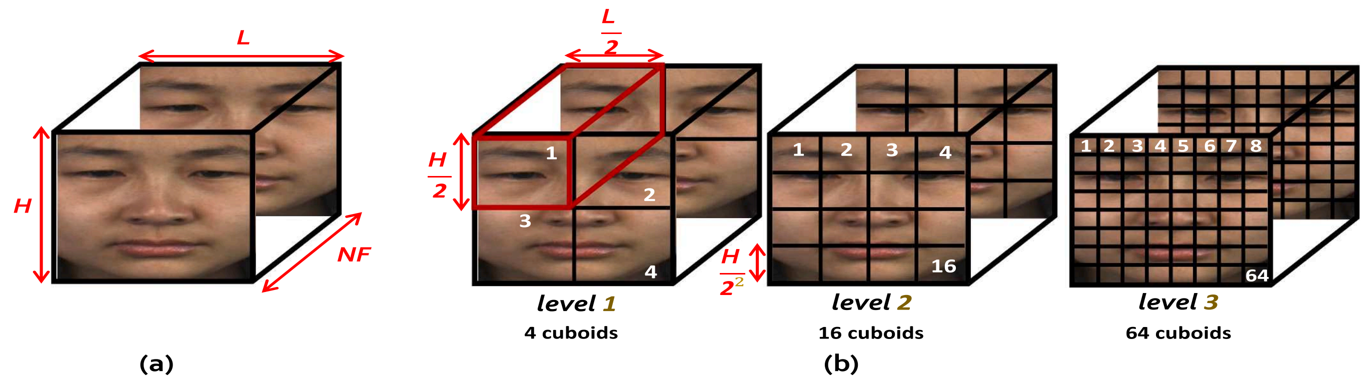


Fig. 2. An example of three levels pyramid representation (a) 3D representation of a video sequence (b) $pLev = [1, 2, 3]$.

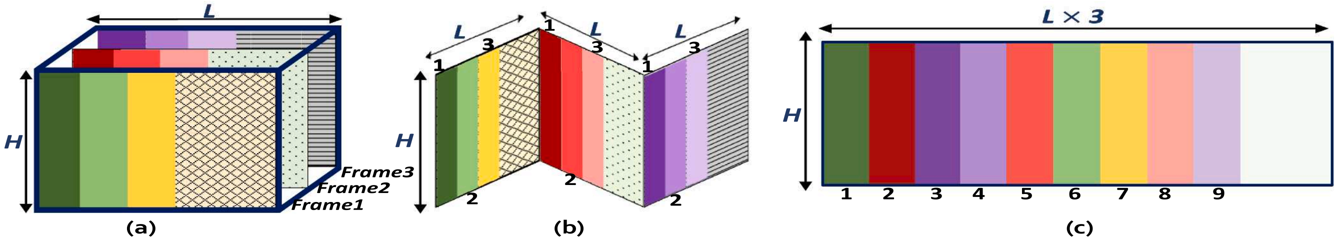


Fig. 3. Accordion transformation (a) Cuboid of a video sequence with 3 frames each of them sized $H \times L$ pixels (b) Accordion representation of frames (c) 2D representation of a cuboid.

(*continued*)

```
Input : $O$: vector of the training video sequences.
        $Sf\{f_1, f_2, ..., f_{nbF}\}$: set of the LBPAccP$^{u2}$ features.
        $trainAccF$: feature matrix of the training videos.
        $nbT$: number of trees.
        $splitNbFeat$: number of features considered to split
        nodes of trees.
begin
    /*Building $forest$ classifier*/
    $[OOB, F^{OOB}, forest] \leftarrow$
      $randomForest(O, trainAccF, nbT, splitNbFeat)$;
    /*Estimating prediction errors $err^{(j)}$ and $err^{(j)}_{f_k}$*/
    for $j$=1 to $nbT$ do
        /*$calculatePredictionError(tree, data)$: this is a function
          that estimates the prediction error of $tree$ using $data$*/
        $err^{(j)} \leftarrow calculatePredictionError(t_j, F^{oob^{(j)}})$);
        for $k$=1 to $nbF$ do
            /*$randomPermute(data, k)$: this is a function that
              permutes randomly the values of $k^{th}$ column of the
              matrix $data$*/
            $p\_F \leftarrow randomPermute(F^{oob^{(j)}}, k)$;
            $err^{(j)}_{f_k} \leftarrow calculatePredictionError(t_j, p\_F)$;
    /*Calculating the importance score of each feature*/
    $score \leftarrow zeros(nbF)$;
    for $k$=1 to $nbF$ do
        $score(k) \leftarrow \frac{\sum_{j=1}^{nbT} err^{(j)} - err^{(j)}_{f_k}}{nbT}$
    /*Determining the most discriminating features*/
    $SfSorted \leftarrow sortSetDesc(Sf, score)$; /*Organizing $Sf$
      in descending order according to $score$*/
    /*$selectBestFeatures(sSet, \Upsilon)$: this is a function that returns the
      set of the most discriminating features from the sorted set $sSet$
      providing the best RF classifier, generated by the feature matrix
      $\Upsilon$, via the accuracy rate*/
    $selectedSf \leftarrow selectBestFeatures(SfSorted, trainAccF)$;
```

Given O the vector of the training video sequences and *trainAccF* its corresponding LBPAccP^{u2} feature matrix, four main steps are required to select the most discriminating features for facial micro-expression recognition:

- The first step focuses on generating the classifier *forest* with *nbT* trees t_j ($j = 1..nbT$), the set out-of-bag of each tree $oob^{(j)}$ and its corresponding feature matrix $F^{oob^{(j)}}$. This is performed by applying the function $randomForest(O, trainAccF, nbT, splitNbFeat)$ where *splitNbFeat* is the number of features selected randomly from F on each split of node to build a tree.
- The second step aims at calculating an importance score for each feature from the set *Sf*, referred to as f_k, with $k \in [1..nbF]$. Precisely, for each tree t_j, we estimate the prediction error of t_j of its corresponding feature matrix $F^{oob^{(j)}}$, noted $err^{(j)}$. This error is defined as $1 - \tau\_oob^{(j)}$. After that, we permute randomly the value of the first column of $F^{oob^{(j)}}$ and we estimate the prediction error of t_j again on the modified feature matrix. The permutation operation is repeated *nbF* times and performed on all the columns of $F^{oob^{(j)}}$ one by one. Therefore, for each tree t_j, we would obtain *nbF* prediction errors, noted $err^{(j)}_{f_k}$, corresponding to the *nbF* operations of permutation. Further, we calculate the difference between $err^{(j)}$ and each $err^{(j)}_{f_k}$ prediction errors to determine the importance scores.
- The third step consists of organizing the set *Sf* according to the obtained importance score. The result is saved in the sorted set, noted *sortedSf*.
- The fourth step consists of selecting the combination of the most discriminating features that produce the best RF classifier. To this end, firstly, we define $nbF - 1$ combinations of features: the first combination refers to the feature having the first highest importance score, and to the feature having the second one. The second combination refers to the first combination, and to the feature having the third highest importance score, and so on. Then, we generate an RF classifier for each combination of features using the training video sequences. LOSO is adopted as a validation protocol and the accuracy rate as an evaluation metric (Eq. 3). Finally, we consider the best combination that maximizes the accuracy rate with a minimum number of features. This combination represent a set, noted *selectedSf*, which contains *nbV* features where $nbV \ll nbF$.

2.3. Facial micro-expression classification

In order to improve the facial micro-expression recognition performance, we have proposed a classification algorithm *RF_prox* based on RF and proximity measure between video sequences. Its steps are illustrated in Algorithm 3.

Algorithm 3. *RF_prox* for classification

(*continued on next page*)

(*continued*)

```
Input  : $O$: vector of $ntr$ training video sequences.
         $trainAccV$: feature matrix of the training video
sequences.
         $testAccV$: feature vector of a test video
sequence.
         $classes$: vector of facial expressions.
         $trainIClasses$: vector of class indices of the
training video
         sequences.
         $nbT$: number of trees.
         $splitNbFeat$: number of features considered to
split nodes of trees.
begin
    /*Building $forest$ classifier*/
    $[OOB, F^{OOB}, forest] \leftarrow$
      $randomForest(O, trainAccV, nbT, splitNbFeat)$;
    /*Estimating prediction errors $err^{(j)}$ and $err^{(j)}_{f_k}$*/
    $prox \leftarrow zeros(ntr)$;
    for $vid$=1 to $ntr$ do
        for $j$=1 to $nbT$ do
            /*$determineIndexLeaf(\Theta, tree)$: this is a function that
              determines the index of leaf node of $tree$ using the
              feature vector $\Theta$*/
            $lTr(vid, j) \leftarrow$
              $determineIndexLeaf(trainAccV(vid, :), t_j)$;
            $lTe(j) \leftarrow$
              $determineIndexLeaf(testAccV, t_j)$;
            /*Computing the proximity measures between the test
              video sequence and each training video sequence*/
            if ($lTr(vid, j)$=$lTe(j)$) then
                $prox(vid) + +$;
    /*Predicting the class of the test video sequence*/
    $classTestVideo \leftarrow$
      $preditClass(prox, trainAccV, trainIClasses, classes)$;
```

RF_prox is based principally on computing *ntr* proximity measures between a test video sequence and the *ntr* training video sequences. The proximity measure between a test video sequence and a training video sequence is defined as the percentage of trees in the RF classifier *forest* that use the same path to predict the class of the training video sequence and the test one.

Formally, given that *trainAccV* is the LBPAccP^{u2} feature matrix (after selecting the most discriminating features) of the *ntr* training video sequences; *testAccV* is the LBPAccP^{u2} feature vector (after selecting the most discriminating features) of the test video sequence; and *trainIClasses* is a vector of *ntr* rows, which represent the class indices of the training video sequences. We have defined five principal steps to predict the facial expression class of the test video sequence:

- In the first step, we build the RF classifier, referred to as *forest*, by applying the standard RF algorithm [44] as explained in the previous section.
- In the second step, we determine, for each training video sequence, the index of the leaf of each tree in *forest*. The result of this step is a matrix, noted *lTr*, of *ntr* rows representing the *ntr* training video sequences and *nbT* columns representing the number of trees used for building *forest*.
- In the third step, we determine, for the test video sequence, the index of the leaf of each tree in *forest*. The result of this step is a vector of *nbT* columns, noted *lTe*.
- In the fourth step, we calculate the proximity measures between the test video sequence and each training video sequence. The result of this step is a 1-dimensional vector of *ntr* columns, noted *prox*.

  Formally, using the Kronecker delta function [48] defined by Equation-System 1, we define the proximity measure between a test video sequence $video_{test}$ and a training video sequence $video_{training}$, referred to as $proximity(video_{test}, video_{training})$, as the number of trees where the same leaf node was used to predict the class of the training and the test videos (cf. Eq. 2):

$$\delta_{AB} = \begin{cases} 0, & \text{if } A \neq B \\ 1, & \text{if } A = B \end{cases} \tag{1}$$

$$proximity(video_{test}, video_{training}) = \sum_{j=1}^{nbT} \delta_{leaf(video_{test}, j) leaf(video_{training}, j)} \tag{2}$$

  where $leaf(X, j)$ represents the index of the leaf of the tree j used to predict the class of X.
- In the fifth step, we predict the class of the test video sequence based on its vector of proximity measures *prox*. This step is defined by the function $PredictClass(prox, trainAccV, trainIClasses, classes)$, presented in Algorithm 4. More specifically, a test video sequence is considered similar to a training video sequence if its corresponding proximity measure corresponds to the maximum value calculated from the vector *prox*. The class of the test video sequence is just one of the *nbCl* facial expressions, represented by the vector *classes*, which has the maximum of scores, referred to as by the vector *expressionsscores*. The score of the i^{th} facial expression $expressionsscores(i)$ is defined as the number of the similar training video sequences corresponding to the maximum value from *prox* divided by the total number of the training video sequences that are labeled by $classes(i)$, the i^{th} facial expression. As a result, the class of the test video sequence is the class that has the maximum score from *expressionsscores*. If more than one facial expression has the maximum score, we consider one of them randomly. It is important to mention that considering facial expressions' scores instead of facial expressions' number to avoid the impact of the imbalance of the different facial expressions on the classification.

Al-
gorithm 4Predicting facial micro-expression

(*continued on next page*)

(*continued*)

```
Function
 $predictClass(prox, trainAccV, trainIClasses, classes)$
   Output: $classTestVideo$: facial micro-expression
           class of the test video sequence to predict.
   begin
       $nbCl \leftarrow size(classes)$; /*$nbCl$: number of classes*/
       $[ntr, nbV] \leftarrow size(trainAccV)$; /*$ntr$: number of
         training video sequences ; $nbV$: number of features*/
       /*Computing maximum of proximity measures*/
       $maxProx \leftarrow max(prox)$;
       /*Initializing $expressionsVideos$, the vector that contains the
         number of video sequences per facial expression*/
       $expressionsVideos \leftarrow zeros(nbCl)$;
       /*Initializing $expressionsMaxProx$, the vector that contains
         the number of similar training video sequences
         corresponding to $maxProx$ per facial expression*/
       $expressionsMaxProx \leftarrow zeros(nbCl)$;
       /*Initializing $expressionsScore$, the vector that contains the
         score of predicting each facial expression*/
       $expressionsScore \leftarrow zeros(nbCl)$;
       for $vid$=1 to $n$ do
           /*$searchIndex(el, \Theta)$: this is a function that returns the
             index of element $el$ in the vector $\Theta$*/
           $indexExpression \leftarrow$
             $searchIndex(trainIClasses(vid), classes)$;
             $expressionsVideos(indexExpression) + +$;
           if $(prox(vid) = maxProx)$ then
               $expressionsMaxProx(indexExpression) + +$;
       for $class$=1 to $nbCl$ do
           $expressionsScore(class) \leftarrow$
             $\frac{expressionsMaxProx(class)}{expressionsVideos(class)}$;
       /*Extracting the maximum from $expressionsScore$*/
       $maxClass \leftarrow max(expressionsScore)$;
       /*Predicting class*/
       /*$searchIndexes(el, \Theta)$: this is a function that returns a
         vector of indices of element $el$ in the vector $\Theta$*/
       $indicesMaxClass \leftarrow$
         $searchIndexes(maxClass, expressionsScore)$;
       /*$randomSelect(\Theta)$: this is a function that selects randomly
         an element from the vector $\Theta$*/
       $indexTestExpression \leftarrow randomSelect(indicesMaxClass)$;
       $classTestVideo \leftarrow classes(indexTestExpression)$;
```

To further explain the *RF_prox* classifier, we consider the following example. Let's consider that we have 12 video sequences labelled $C1$,$C2$, and $C3$ that will be used as our training dataset and 3 video sequences are our test dataset.

Using the *RF* algorithm, we build a forest of 4 trees (cf. Fig. 4). The number of leaves of these trees is variable. Each leaf has an index. This forest will be used to extract the index of the leaf of each tree (noted `Leaf_Index(`*Video*`, treei_i = 1..4 )` to predict the class of the video sequence. So we will obtain a matrix of 15 rows (training and test video sequences) and 5 columns (4 leaf indexes and the class) (see Table 1).

From the Table 1, we will build an occurrence matrix of 3 rows that correspond to the test video sequences and 12 columns that correspond to the training datasets (cf. Table 2). Each value of this matrix represents the number of trees that use the same leaf index to predict the class of the test and training video sequences. For example, when we compare test video 1 and the training video 1, we can find only one tree (tree number 3) that use the same index leaf (index 3) to predict the class of the video sequences. The possible values of this matrix vary between 0, that represents the maximum of the dissimilarity, and 4 that represents the maximum of similarity.

For each test video sequence, we select the training video sequences that achieve the maximum of proximities. From these selected training video sequences, we count the number of sequences for each class (cf. Table 3). For example, the maximum of the proximity of the test video 1 equals 4 corresponding to three training video sequences (Training video 5, Training video 10, and Training video 12): 1 is labeled C1 and 2 are labeled C3. In the training dataset, the total of the training sequences labeled C1 (resp. C3) is 6 (resp. 3). Based on these values, a score equals 1/6 will be assigned to C1, a score equals 0 will be assigned to C2, and a score equals 2/3 will be assigned to C3. So, the predicted class of the test video 1 is C3 because the score of C3 is greater than that of C1. Similarly,

Table 1
Leaf Indexes of the training and test video sequences.

Video	Leaf Index				Class
	Tree 1	Tree 2	Tree 3	Tree 4	
Training 1	2	1	3	1	C1
Training 2	2	4	1	4	C1
Training 3	2	1	1	2	C1
Training 4	1	4	3	2	C1
Training 5	1	2	3	4	C1
Training 6	1	3	2	2	C1
Training 7	2	4	3	4	C2
Training 8	2	1	3	1	C2
Training 9	3	1	2	2	C2
Training 10	1	2	3	4	C3
Training 11	1	2	3	3	C3
Training 12	1	2	3	4	C3
Test 1	1	2	3	4	?
Test 2	2	1	3	1	?
Test 3	2	4	2	2	?

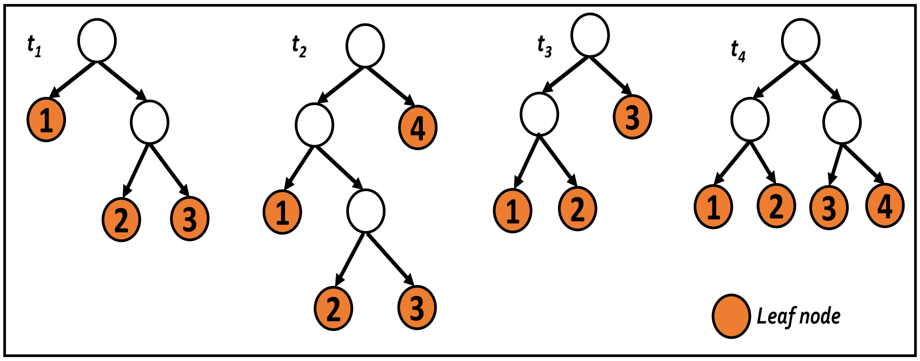


Fig. 4. Forest of four trees.

Table 2
Proximities of the test with the training video sequences.

Test Video	Training Video											
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Test 1	1	1	0	2	4	1	2	1	0	4	3	4
Test 2	4	1	2	1	1	0	2	4	1	1	1	1
Test 3	1	2	2	2	0	2	2	1	2	0	0	0

Table 3
Prediction of the test video sequences classes.

	Test video 1			Test video 2			Test video 3		
Max Prox	4			4			2		
Classes	C1	C2	C3	C1	C2	C3	C1	C2	C3
Occurence Max Prox	1	0	**2**	**1**	**1**	0	**4**	2	0
Score	1/6	0/3	**2/3**	1/6	**1/3**	0/3	**4/6**	**2/3**	0/3
Prediction	C3			C2			C2		

the predicted class of the test video 2 is C2. For the test video 3, the classes C1 and C2 have the same score. Therefore, the predicted class of the test video 3 can be C1 or C2. In our solution, we randomly opt for C1 or C2 as the class of the test video 3.

3. Experimental framework

This section is organized into three sub-sections. In SubSection 3.1, we present the different used datasets of facial micro-expressions. In SubSection 3.2, we study the choice of LBPAccP^{u2} and RF hyper-parameters. In SubSection 3.3, we define the experimental setup and the evaluation metrics.

3.1. Datasets

Due to the difficulty of capture spontaneous and even posed facial micro-expressions, very few facial micro-expression datasets are available. The most famous datasets include SMIC [49], CasmeI [50], CasmeII [13], Cas(me)^2 [16], and MEVIEW [51]. They present different types of facial micro-expressions.

SMIC dataset [49] consists of 164 facial micro-expression video sequences of 16 subjects captured in frontal pose. These sequences are recorded in a controlled scenario using a normal visual camera (VIS) of high speed of 100fps with a resolution of 640 × 480 pixels. All the faces are normalized to a face model [52] and detected according to the eye positions, estimated using a Haar eye detector [53]. In this dataset, three facial micro-expressions are expressed including *"positive"*, *"negative"*, and *"surprise"*, where 51 video sequences are labeled *"positive"*, 70 video sequences *"negative"*, and 43 video sequences *"surprise"*.

CasmeI dataset [50] contains 195 facial micro-expression video sequences of 19 subjects captured in frontal pose. The temporal resolution is 60fps and the spatial resolution for each frame of a video sequence is 640 × 480 pixels. All the faces are manually detected. The video sequences are categorized into seven classes of facial micro-expressions: *"happiness"*, *"repression"*, *"sadness"*, *"disgust"*, *"contempt"*, *"surprise"*, and *"tense"*. The available copy of the CASMEI dataset is slightly different from that described in the original paper [50]. It contains 188 out of 195 video sequences. Using the guide file proposed by the authors, we consider only 165 out of 188 video sequences after excluding the sequences labelled by "sadness" and "contempt". Precisely, 8 video sequences are labeled *"happiness"*, 33 video sequences *"repression"*, 40 video sequences *"disgust"*, 18 video sequences *"surprise"*, and 66 video sequences *"tense"*.

Using the 165 video sequences, we have proposed two versions of Casme I dataset. The first one, called "CasmeI_1", consists of 157 video sequences labeled by *"disgust"*, *"repression"*, *"surprise"*, and *"tense"*. The second one, called "CasmeI_2", consists of 165 video sequences labeled by 8 *"positive"* (corresponding to *"happiness"*), 40 *"negative"* (corresponding to *"disgust"*), 18 *"surprise"*, and 99 *"others"* (corresponding to *"repression"* and *"tense"*). We consider these two versions to be under the same experimental setups as other related works [18,25,26] and to show the impact of generalizing facial micro-expressions into *"positive"*, *"negative"*,*"surprise"* and *"others"*.

CasmeII dataset [13] contains 256 facial micro-expression video sequences of 26 subjects captured in frontal pose. Similar to CasmeI, the spatial resolution for each frame of a video sequence is 640 × 480 pixels. The temporal resolution of video sequences is 200fps. All the faces are normalized to a face model [52] and detected according to the eye positions that are estimated using a Haar eye detector [53]. The facial micro-expressions are distributed as follows: 63 video sequences are labeled *"disgust"*, 27 video sequences *"repression"*, 32 video sequences *"happiness"*, 25 video sequences *"surprise"*, 7 video sequences *"sadness"*, 2 video sequences *"fear"*, and 100 video sequences *"others"*. Like several of the related works [13,21,18,31,15], CasmeII includes 247 out of 256 video sequences. It does not include the video sequences labeled by *"fear"* and *"sadness"* due to their limited number.

Cas(me)^2 dataset [16] consists of 357 video sequences of 22 subjects captured in frontal pose. These video sequences are recorded in a controlled scenario using a high speed of 30fps with a resolution of 640 × 480 pixels. It contains 300 out of 357 video sequences that present facial macro-expressions with low intensity. The rest of the video sequences present facial micro-expressions. These sequences are labeled by four classes: *"positive"*, *"negative"*, *"surprise"*, and *"others"*. All faces are detected via a face model proposed in [54]. The available copy of this dataset is slightly different from that presented in its original paper [16].

Only 345 video sequences are considered: 290 video sequences of facial macro-expressions and 55 video sequences of facial micro-expressions. In our work, we have used two versions of Cas(me)^2. The first one, called "Cas(me)^2_1", consists of 345 video sequences distributed in four classes as follows: 115 video sequences are labeled *"positive"*, 125 video sequences *"negative"*, 23 video sequences *"surprise"*, and 82 video sequences *"others"*. The second version, called "Cas (me)^2_2", contains only 55 video sequences that present facial micro-expressions of 14 subjects: 7 video sequences are labeled *"positive"*, 21 video sequences *"negative"*, 8 video sequences *"surprise"*, and 19 video sequences *"others"*.

3.2. Settings of hyper parameters

The present subsection presents the settings of the hyper parameters of the proposed approach, which include the pyramid levels' combination, the number of trees, and the number of features per split of each node in each tree in RF. These hyper parameters can influence the effectiveness of the proposed approach.

To determine the optimal number of trees *nbT* used for building the RF classifier, we have applied a grid search algorithm of the OOB accuracy rate versus the number of trees for each dataset. The grid search is repeated for four-level combinations of the pyramid $pLev = [1, 2]$, $pLev = [1,3]$,$pLev = [2,3]$, and $pLev = [1,2,3]$. The obtained results are compatible with those shown in [43].

The best performance for each level combination of the pyramid for all the datasets is summarized in Fig. 5. Table 4 depicts the number of trees that corresponds to each best OOB accuracy rate presented in Fig. 5.

According to Fig. 5, the four-level combinations display comparable OOB accuracy rate for all the datasets. The best one is observed by the combination $pLev = [2,3]$. We observe a remarkable improvement in performance when using $pLev = [2,3]$ compared to $pLev = [1,3]$ especially for the SMIC dataset (from 60.98% to 68.89% of OOB accuracy) and CasmeII dataset (from 64.98% to 69.88%). We also observe that

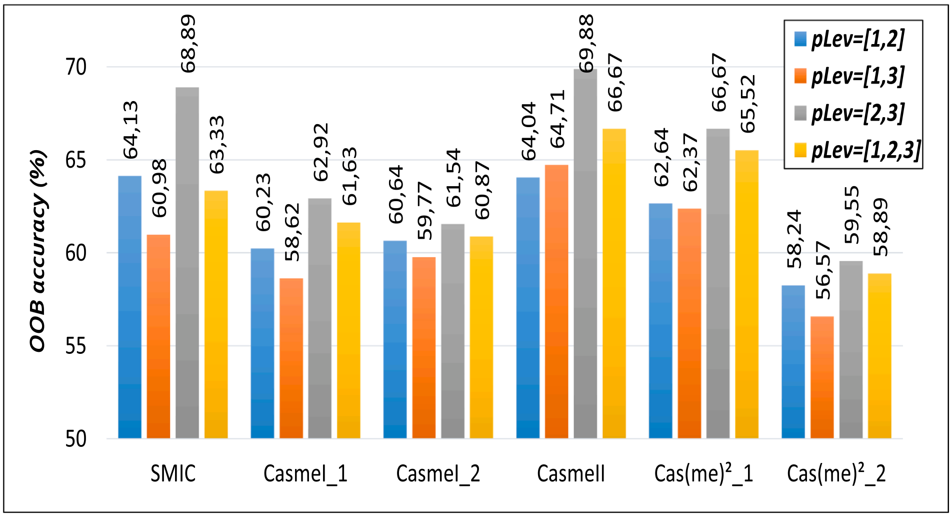


Fig. 5. The best OOB accuracy rate of each RF classifier for the four-level combinations of the pyramid and for each dataset.

Table 4
Number of the considered trees for the four-level combinations of the pyramid for each dataset.

	Number of trees ***nbT***			
	$pLev = [1,2]$	$pLev = [1,3]$	$pLev = [2,3]$	$pLev = [1,2,3]$
SMIC	170	200	**170**	120
CasmeI_1	220	200	**180**	160
CasmeI_2	260	230	**210**	230
CasmeII	230	250	**180**	170
Cas(me)^2_1	210	130	**170**	280
Cas(me)^2_2	250	240	**230**	260

using $pLev = [2,3]$ shows a slight improvement in performance relative to $pLev = [1,2,3]$ for all the datasets.

The right choice of the number of trees *nbT* leads to improve the performance and the spatial complexity of the RF classifier [55]. Increasing the number of trees would bring about no significant performance gain and would only increase the computational cost. As illustrated in Table 4, the adequate number of trees does not exceed 280 trees regardless of the dataset, and the pyramid levels' combination. We observe that the adequate number of trees based on the LBPAccP^{u2} feature space using $pLev = [2,3]$ is not always the lowest one when we compare it with the number of trees of the other level combinations. Despite the slight increase in the computational cost of the forest building, using $pLev = [2,3]$ increases remarkably the performance of the proposed approach.

To conclude, in the subsequent experiments, we consider the sub-regions of level 2 and level 3 to compute the LBPAccP^{u2} feature space and the adequate number of trees corresponding to this space to generate the RF classifier.

3.3. Experimental setup

To evaluate the proposed approach, we have performed two fundamental experimental series:

Series 1: Evaluating the impact of the feature selection on the performance of our approach;

Series 2: Evaluating the performance of the proposed approach.

For all experiments, we used the Leave-One-Subject-Out (LOSO) cross validation protocol, where the video sequences from one subject were used for testing and the rest for training. This task was repeated until each subject was used once as the test set.

As a primarily validation metric, we have calculated the accuracy rate, noted τ, defined as follows in Eq. 3:

$$\tau = \frac{\sum_{f=1}^{m}\sum_{cl=1}^{nbCl} v_f^{cl}}{v} \tag{3}$$

where m is the number of subjects in the dataset; v is the total number of video sequences in the dataset; and $v_f^{cl}$is the number of video sequences of the f^{th} subject correctly classified as the class cl ($cl \in [1..nbCl]$).

The majority of the datasets present an imbalanced distribution of facial micro-expressions. Therefore, as the accuracy rate favors classes with the larger number of video sequences over classes with smaller ones, this metric is not suitable to evaluate the performance of the proposed approach [56]. As recommended in [57], we have used the F-measure metric (Eq. 4), as the second evaluation metric, to overcome this issue.

$$F - measure = \frac{2 \times precision \times recall}{precision + recall} \tag{4}$$

where $precision = \frac{\sum_{cl=1}^{C} precision^{cl}}{C}; precision^{cl} = \frac{\sum_{f=1}^{m} v_f^{cl}}{\sum_{f=1}^{m}\sum_{i=1}^{C} v_f^{i,cl}}$; $recall = \frac{\sum_{cl=1}^{C} recall^{cl}}{C}; recall^{cl} = \frac{\sum_{f=1}^{m} v_f^{cl}}{\sum_{f=1}^{m}\sum_{i=1}^{C} v_f^{cl,i}}$; v_f^{cl} denotes the number of video sequences of the f^{th} subject correctly classified as the class cl; and $v_f^{a,b}$is the number of video sequences of the f^{th} subject labeled by the class "a" and predicted as the class "b".

4. Results and discussions

This section is devoted to discuss the evaluation of the proposed approach.

4.1. Evaluation of the impact of the feature selection

The goal of the first experimental series is to study how well the feature selection algorithm generalizes on the proposed approach. Using the 4720 features extracted from the levels 2 and 3 of the pyramid representation of LBPAccP^{u2}, our goal is to select *nbV* most discriminating features with *nbV* that should be significantly lower than 4720. The number of the most discriminating features for each dataset is summarized in Table 5. It depends on the size and the nature of the dataset and varies from 76 to 91.

Fig. 6 compares the performance of the proposed solution with and

Table 5
Number of the most discriminating LBPAccP^{u2} features per dataset.

	Number of features
SMIC	86
CasmeI_1	91
CasmeI_2	88
CasmeII	79
Cas(me)^2_1	76
Cas(me)^2_2	82

without feature selection according to the accuracy rate for the six datasets.

The obtained results prove the effectiveness of the used feature selection algorithm. Indeed, for all the datasets, the use of the feature selection algorithm increases the performance of the classifier. This enhancement varies from one dataset to another and reaches 4.86% of accuracy and 6% of F-measure for the CasmeII dataset. Furthermore, using a PC with Intel(R) Core i5-8250U CPU @ 1.60 GHz 1.80 GHz and 8 GB of Random Access Memory (RAM), the computation time of features before selection decreases from 2.3s to 1.1s after selection.

Fig. 7 details the results of the F-measure per facial expression for each dataset. As shown in this figure, the rare facial expressions of datasets (i.e. the facial that include a limited number of samples),

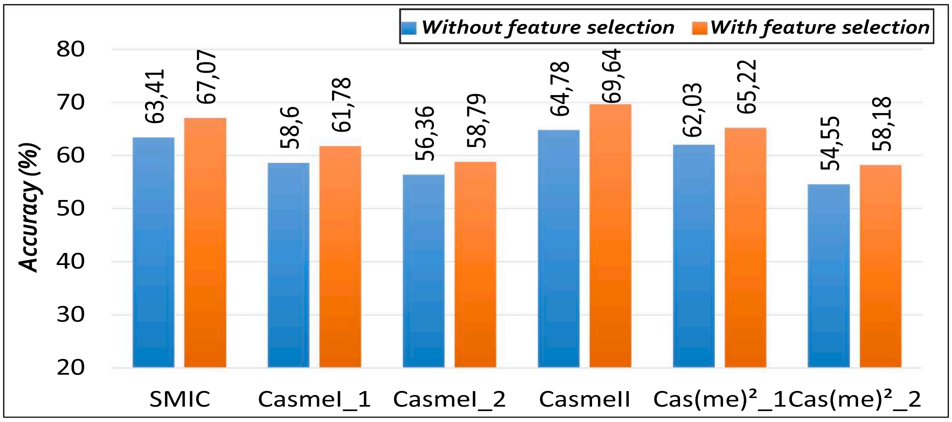


Fig. 6. Impact of feature selection algorithm on the performance of the classifier according to the accuracy rate.

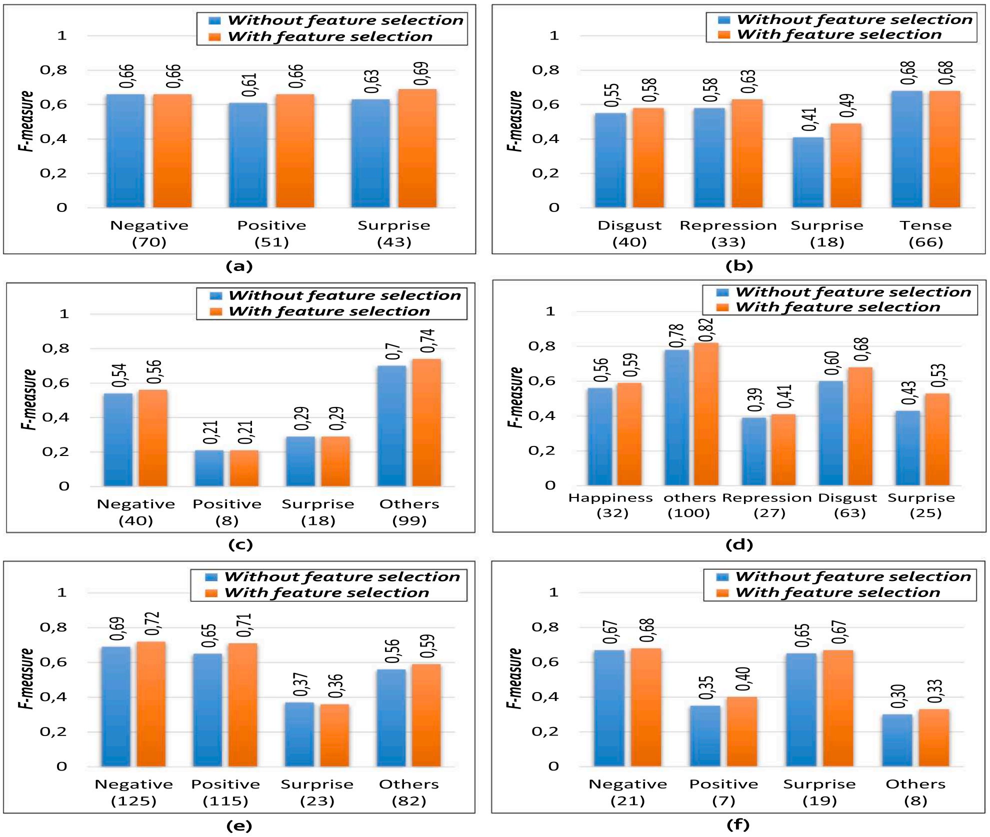


Fig. 7. F-measure per facial expression of the classifiers generated by the RF (with and without feature selection) on all the datasets (a) SMIC (b) CasmeI_1 (c) CasmeI_2 (d) CasmeII (e) Cas(me)^2_1 (f) Cas(me)^2_2.

display the lowest performances regardless of the use of the feature selection algorithm.

In most of the cases, the selection of the most discriminating features improves the performance of the classifier. This improvement is not fixed and depends on the facial micro-expression and the dataset. The best enhancement is shown on the recognition of *"surprise"* of SMIC (6%), CasmeI_1 (8%), and CasmeII (10%); *"positive"* of Cas(me)^2_1 (6%) and Cas(me)^2_2 (5%); and *"others"* of CasmeI_1 (4%). For some facial micro-expressions like *"negative"* of SMIC, *"tense"* of CasmeI_1, and *"positive"* and *"surprise"* of CasmeI_2, the selection of the discriminating features does not have any effect. As a result, when the original space is used without feature selection, only the features produced by the feature selection is used to build the RF classifier.

To sum up the results, we conclude that the use of feature selection algorithm, which is based on RF, can significantly improve the performance of facial micro-expression recognition and remarkably decrease the size of the feature space.

4.2. Evaluation of the proposed approach

To prove the effectiveness of the proposed approach for facial micro-expression recognition, we perform a comprehensive comparison of all datasets between the classifiers generated by the standard RF algorithm and the proposed *RF_prox* algorithm using the LBPAccP^{u2} after selection. Fig. 8 compares the performance of RF and *RF_prox* via the accuracy rate for the six datasets.

As shown in this figure, we conclude that the use of *RF_prox* improves the accuracy rate. The enhancement of accuracy rate depends on the dataset. It varies from 7.27% (Cas(me)^2_2) to 11.74% (CasmeII). For the rest of the datasets, the enhancement roughly equals 10%. Thanks to the *RF_prox* algorithm, the accuracy rate exceeds 65% and may reach 81.38%. To the authors' best knowledge, the obtained results represent one of the best findings achieved in facial micro-expression recognition research fields. This proves the effectiveness of the proposed *RF_prox* classification algorithm. The F-measure per facial expression are displayed in Fig. 9 for all the datasets. Comparing to the standard RF algorithm, *RF_prox* shows its efficiency in improving the performance of recognizing all kinds of facial expressions for all the datasets.

More concretely, for SMIC, the F-measure enhancement of *"negative"*, *"positive"*, and *"surprise"* equals 12%, 11%, and 6%, respectively. Furthermore, for CasmeI_1, the F-measure enhancement of *"disgust"*, *"repression"*, and *"tense"* is more obvious than that of "surprise". For CasmeI_2, the F-measure enhancement varies from 8% (*"surprise"*) to 12% (*"negative"*). The classifier of CasmeI_1 is more accurate than that of CasmeI_2 in the recognition of *"surprise"*. As a matter of fact, the *"surprise"* F-measure decreases from 53% to 37%. This can be explained that combining the *"disgust"*, the *"repression"*, and the *"tense"* classes of CasmeI_1 to create new classes in CasmeI_2 (*"negative"*, *"positive"*, and *"others"*) can be at the origin of the misclassification of *"surprise"* in CasmeI_2. Moreover, for CasmeII, the best F-measure improvement is showed in the recognition of *"happiness"* (18%) and *"repression"* (19%). For Cas(me)^2_1, the best F-measure improvement is showed in the recognition of *"surprise"* (15%). Unlike Cas(me)^2_1, the *"surprise"* facial micro-expression in Cas(me)^2_2 shows the worst F-measure enhancement i.e. its corresponding RF classifier presents already a good performance.

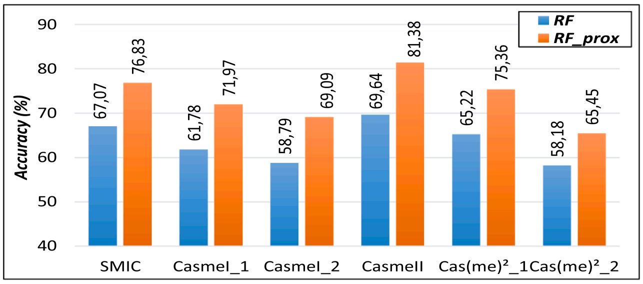


Fig. 8. Accuracy results of the standard RF algorithm versus the *RF_prox* algorithm.

Considering all the datasets, in addition to the best performance in the recognition of most of the facial expressions, the proposed approach considerably improves the recognition of rare ones. For CasmeI_2 and Cas(me)^2_2, this improvement reaches 10% and 17% of *"positive"*, respectively. For CasmeII and Cas(me)^2_1, the F-measure improvement reaches 14% and 15% of *"surprise"*, respectively.

4.3. Statistical tests for performance comparison

In the previous experiments, we observe that the use of the standard RF algorithm with feature selection improves the results compared to the use of that without feature selection in terms of the accuracy rate and the F-measure for facial micro-expression recognition. Furthermore, we observe that the use of the proposed *RF_prox* algorithm with feature selection shows better performance than that provided by the standard RF algorithm with feature selection. So, to explore whether the achieved enhancement with the feature selection and the proximity measures in RF is significant or not, we have performed the Friedman Aligned ranks test. Three classifiers are distinguished: (1) *RF_prox* with feature selection, (2) RF with feature selection and (3) RF without feature selection according to the F-measure on the six datasets.

More specifically, we opted for Friedman aligned ranks [58] as the number of approaches is less than 4. Indeed, according to the STAC Web platform exploited for the generation of the statistical tests [59], we should use Friedman aligned ranks when the number of classifiers is less than 4. If the statistical test rejected the null hypothesis affirming the similarity of means of two or more algorithms, the Holm post hoc test should be performed in order to find out which algorithm rejects the equality hypothesis with respect to a selected control method. Therefore, the adjusted $p-value$ associated with each comparison was computed, which represents the lowest level of significance of a hypothesis that results in a rejection [60]. We fixed the level significance α to 0.05 for all the statistical tests. Table 6 reports the Friedman Aligned ranks test of three classifiers.

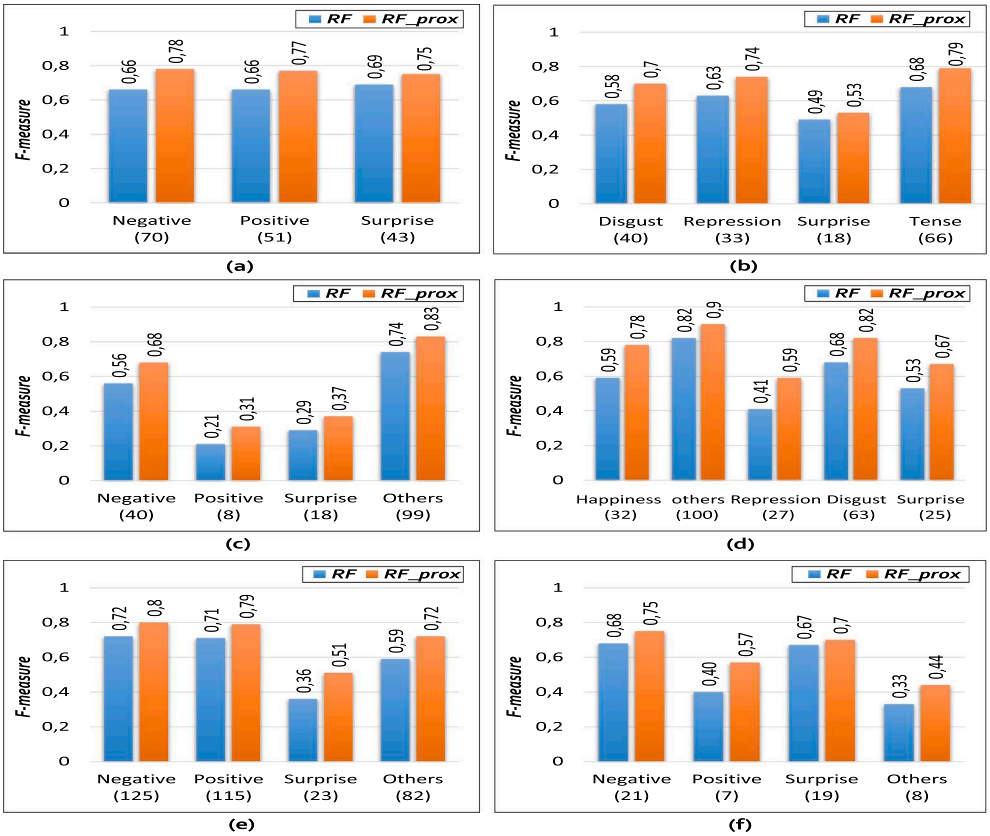
Obviously, the best classifier is the *RF_prox* with feature selection, which is shown the lowest ranks. Therefore, we consider this classifier as the control method.

The Friedman Aligned Ranks test shows a $p-value$ equals 0.00452. Thus, it rejects the null hypothesis. Table 7 displays the comparative statistical results using the Friedman Aligned to rank each classifier along with the adjusted $p-value$ obtained by means of a Holm test to conclude whether the classifiers perform similarly regarding a significance level $\alpha = 0.05$.

We conclude that the enhancement of *RF_prox* with feature selection versus the RF without feature selection is significant. This affirms the adequacy of computing the proximity measures in RF. Compared to RF with feature selection, *RF_prox* does not show significant enhancement. As a result, this proves the effectiveness of the feature selection algorithm.

4.4. Impact of the random parameters on the effectiveness of the proposed approach

Considering that RF is based on a random selection of samples and features, we decide to repeat the experiments based on 10 times in order to evaluate the impact of the random selection on the proposed approach. For each iteration k ($k \in [1..10]$), we calculate the accuracy rate, noted τ_k. Then, we measure the standard deviation (Eq. 5), noted σ, in order to quantify the magnitude of the dispersion of τ_k from their mean, noted M. So, a low σ indicates that random selection does not affect the effectiveness of the proposed approach.



(a) (b) (c) (d) (e) (f)

Fig. 9. The Comparison of the standard RF algorithm vis-á-vis the proposed *RF_prox* algorithm according to F-measure per facial expression on all the datasets (a) SMIC (b) CasmeI_1 (c) CasmeI_2 (d) CasmeII (e) Cas(me)^2_1 (f) Cas(me)^2_2.

Table 6
Friedman Aligned ranks test of *RF_prox* with feature selection, RF with feature selection and RF without feature selection according to the F-measure.

Rank	Algorithm
3.50000	**(1)** *RF_prox* with feature selection
9.50000	**(2)** RF with feature selection
15.50000	**(3)** RF without feature selection

Table 7
Holm test table for the *RF_prox* classifier ($\alpha = 0.05$).

Comparison	Statistic	Adjusted $p-value$	Result
(1) versus **(3)**	3.89331	0.00020	*H0* is rejected
(1) versus **(2)**	1.94666	0.05158	*H0* is accepted

$$\sigma = \sqrt{\frac{\sum_{k=1}^{10} (\tau_k - M)^2}{9}}; \quad M = \frac{\sum_{k=1}^{10} \tau_k}{10} \tag{5}$$

It is worth noting that in all the previous experiments based on RF, we consider the results of the iteration that record an accuracy rate close to the mean *M*.

The average of all the simulation is reported on Table 8.

The standard deviation of all the datasets varies from 1.08 to 2.27. We observe that the effectiveness of the proposed classification algorithm does not depend on its randomly chosen parameters.

Table 8
Standard deviation of 10 times evaluation of the *RF_prox* classifiers generated based on LBPAccP^{u2} for all the datasets.

	M(%)±σ(%)
SMIC	76.59±1.16
CasmeI_1	71.66±1.38
CasmeI_2	69.03±1.41
CasmeII	0.81±1.08
Cas(me)^2_1	75.65±1.34
Cas(me)^2_2	65.45±2.27

5. Comparative study

To analyze the performance of the proposed approach, we compare it on the one hand with several micro-expression recognition approaches in the literature using SMIC, CamseI, CamseII, and Cas(me)^2. This comparison is divided into two parts: (i) comparison with the approaches that share the same experimental setups; and (ii) comparison with the approaches not sharing the same experimental setups. On the other hand, we evaluate the proposed approach using cross datasets. Indeed, the classifier must be able to classify facial micro-expressions of a subject who does not belong to the same capture environment. This experiment simulates the real life situation when the system is used to recognize facial micro-expressions on the unseen data (different resolutions, color depth, ethnicity, gender, age).

5.1. Comparison with approaches sharing the same experimental setups

We compare the proposed approach with some related works using

SMIC and CasmeII. Indeed, they consider 164 (resp. 247) samples from the SMIC (resp. CasmeII) dataset, which the resolution of video frames equals 170 × 140 (resp. 340 × 280) pixels. The LOSO validation protocol is applied to measure the performance of facial micro-expression recognition.

For SMIC, the facial micro-expressions are classified in three categories: *"positive"*, *"negative"*, and *"surprise"*. For CasmeII, they are classified in 5 categories: *"happiness"*, *"others"*, *"repression"*, *"disgust"*, and *"surprise"*. Tables 9 and 10 report the comparison of our proposal with some related approaches in the literature using SMIC and CasmeII, respectively.

Some of the related works as [13,18,15,19] proposed appearance-based methods using different operators applied to either whole-face or specific sub-regions in the face. For some other works like those of [24–26,28], they based on motion-based methods. Nevertheless, a few works have represented the deep learning-based approaches according to different architectures of deep neural networks as in [31,38].

The first observation that can be made is that the proposed approach based on LBPAccP^{u2} is the best one according to the LOSO accuracy rate reaching 76.83% and 81.38%, using SMIC and CasmeII, respectively.

The methods discussed in [13,21,18,15,20] are directly comparable to our proposal. Indeed, in these methods, the authors proposed an appearance spatio-temporal feature space based on LBP [45]. The enhancement of the accuracy rate varies from 5.51% to 22.59%. Our proposal also outperforms the other works [24–26,28,27], which shows its ability to analyze the Optical Flow in preserving motion changes. The enhancement of the accuracy rate varies from 2.67% to 34.88%.

Other related works as [15,28] include the motion magnification as a preprocessing step. These works achieve the best performances in the literature, but they are less efficient than that of the proposed approach. In other words, even if we have not used a preprocessing step as the motion magnification, our approach still shows better results. Obviously, more experiments will be needed to verify if the incorporation of a preprocessing step as the motion magnification can improve the effectiveness of our proposal.

The strength of our solution lies in the low size of its feature space. As a matter of fact, when we compare it with the size of the feature space of related works, the difference is more than significant. Contrary to the related works [13,21,18], our solution needs 86 features on SMIC and only 79 features on CasmeII to well recognize facial micro-expressions. So, using a few discriminating features is enough to create a strong classifier of facial micro-expression recognition that outperforms the famous literature approaches that are based on a high-dimensional feature space.

Table 9
The comparison of the proposed approach vis-a-vis the related approaches on the SMIC dataset.

	Feature space	Class. Algo[a]	Nb. Samples[b]	Nb. Features[c]	τ (%)[d]
Huang et al., 2015 [21]	STLBP-IP	SVM	164	4956	57.93
Huang et al., 2016 [18]	STCLQP	SVM	164	23040	64.02
Xu et al., 2017 [25]	FDM	SVM	164	–	54.88
Liong et al., 2018 [20]	LBP-TOP (RoI)	SVM	163	–	54.00
Liong et al., 2018 [27]	Bi-WOOF	SVM	164	288	62.20
Lu et al., 2018 [26]	FMBH	SVM	164	288	71.95
Zong et al., 2018 [19]	STLBP-IP	KGSL	164	–	60.37
Wang et al., 2019 [29]	Optical Flow	SVM	164	–	71.70
Wang et al., 2020 [39]	ImageNet + Transfer learning◇		164	–	49.40
Ours	**LBPAccP^{u2}**	***RF_prox***	**164**	**86**	**76.59**

[a] *Classification algorithm*
[b] *Number of samples*
[c] *Number of features*
[d] *Accuracy*
◇ Unified process (feature detection + classification).

Table 10
The comparison of the proposed approach vis-a-vis the related approaches on the CasmeII dataset.

	Feature space	Class. Algo[a]	Nb. Samples[b]	Nb. Features[c]	τ (%)[d]
Yan et al., 2014 [13]	LBP-TOP	SVM	246	4425	63.41
Huang et al., 2015 [21]	STLBP	SVM	247	4956	59.51
Liu et al., 2015 [24]	MDMO	SVM	246	72	67.37
Huang et al., 2016 [18]	STCLQP	SVM	247	38440	58.30
Kim et al., 2016 [31]	CNN + LSTM architecture◇		246	–	60.98
Wang et al., 2017 [15]	LBP-TOP *magnified*	SVM	247	4425	75.30
Xu et al., 2017 [25]	FDM	SVM	246	–	45.93
Zheng et al., 2017 [10]	2DGSR	SVM	236	–	64.88
Liong et al., 2018 [27]	Bi-WOOF	SVM	246	288	58.85
Lu et al., 2018 [26]	FMBH	SVM	246	288	69.11
Li et al., 2018 [28]	HIGO *magnified*	LSVM	247	1024	78.14
Wang et al., 2019 [29]	Optical Flow	SVM	246	–	69.56
Zhi et al., 2019 [38]	3D-CNN + Transfer learning LSVM		247	256	65.99
Abdallah et al., 2019 [42]	PCA [PTLBP^{u2}]	RF	247	83	65.79
Ours	**LBPAccP^{u2}**	***RF_prox***	**247**	**79**	**80.81**

[a] *Classification algorithm*
[b] *Number of samples*
[c] *Number of features*
[d] *Accuracy*
◇ Unified process (feature detection + classification)

Concerning the used classification algorithm, the majority of the approaches are based on a single technique, in particular SVM, but our approach is based on an ensemble learning method (i.e. RF).

5.2. *Positioning with approaches not sharing the same experimental setups*

The literature approaches that use CasmeI and Cas(me)^2 do not share the same experimental setups. Indeed, the frame resolution, the number of samples and the facial micro-expressions to be classified vary from one approach to another. It is difficult to make a quantitative comparison relative to our approach. Under the LOSO validation protocol, we just positioned our proposal in terms of the number of samples, the number of features, and the accuracy rate. The results of using CasmeI and Cas(me)^2 are summarized in Tables 11 and 12, respectively.

For CasmeI_1, all the approaches are based on the SVM algorithm. The proposed approach, based on the *RF_prox* algorithm, displays a lower number of features equals 91 and better performance in terms of the accuracy rate. It can outperform the other related works that consider a superior number of samples [18,25] as well as an inferior number of samples [26].

For CasmeI_2, compared to the study of [24], the proposed approach records a slight increase of 0.17% with a similar number of features. This affirms the importance of selecting discriminating features. Compared to

Table 11
The positioning of the proposed approach vis-a-vis the related approaches on the CasmeI dataset.

	Feature space	Class. Algo[a]	Nb. Samples[b]	Nb. Features[c]	τ (%)[d]
CasmeI_1					
Huang et al., [18]	STCLQP	SVM	171	23040	57.31
Xu et al., 2017 [25]	FDM	SVM	171	–	56.14
Lu et al., 2018 [26]	FMBH	SVM	150	–	61.33
Ours	**LBPAccP^{u2}**	*RF_prox*	**159**	**91**	**71.66**
CasmeI_2					
Liu et al., 2015 [24]	MDMO	SVM	167	72	68.86
Zheng et al., 2017 [10]	2DGSR	SVM	97	–	71.19
Ours	**LBPAccP^{u2}**	***RF_prox***	**165**	**88**	**69.03**

[a] *Classification algorithm*
[b] *Number of samples*
[c] *Number of features*
[d] *Accuracy*

Table 12
The positioning of the proposed approach vis-a-vis the related approaches on the Cas(me)^2 dataset.

	Feature space	Class. Algo[a]	Nb. Samples[b]	Nb. Features[c]	τ (%)[d]
Cas(me)^2_1					
Qu et al., 2018 [16]	LBP-TOP	SVM	356	12288	40.95
Lu et al., 2018 [26]	FMBH	SVM	357	–	73.67
Ours	**LBPAccP^{u2}**	*RF_prox*	**345**	**76**	**75.65**
Cas(me)^2_2					
Liong et al., 2018 [27]	Bi-WOOF	SVM	54	288	59.26
Ours	**LBPAccP^{u2}**	*RF_prox*	**55**	**82**	**65.45**

[a] *Classification algorithm*
[b] *Number of samples*
[c] *Number of features*
[d] *Accuracy*

[10], the performance of the proposed approach presents a decrease of about 2%. This can be explained that Zheng [10] used only 97 samples to evaluate his approach.

Using the facial micro and macro-expression samples of Cas(me)^2_1, our proposal achieves very good performance in terms of the accuracy rate and the number of features. Furthermore, using only the facial micro-expression samples of Cas(me)^2_2, the proposed approach shows better performance than [27].

5.3. *Positioning the approach proposed using cross datasets*

In this section, we used a another dataset, named MEVIEW dataset [51], that consists of 27 facial micro-expression video sequences captured under uncontrolled environment (in the Wild). They are collected from poker games and TV interviews downloaded from the Internet. The advantage of poker games is the stress factor and the need to hide emotions. Indeed, players try to conceal or fake their true emotions, which is a scenario where facial micro-expressions are likely to appear. In this dataset, five facial micro-expressions are expressed including 7 *"contempt"*, 3 *"fear"*, 2 *"angry"*, 9 *"surprise"*, and 6 *"happiness"*. In our work, we distributed the videos sequences as follows: 12 labeled by *"negative"* (corresponding to *"contempt"*, *"fear"*, and *"angry"*), 6 labeled by *"positive"* (corresponding to *"happiness"*), and 9 labeled by *"surprise"*.

The evaluation of the proposed approach using cross datasets is performed in five different scenarios: the first one considers CasmeII as a learning dataset, and SMIC as a test dataset; the second one considers SMIC as a learning dataset, and CASMEII as a test dataset; the third one considers CasmeII as a learning dataset, and MEVIEW as a test dataset; the fourth one considers SMIC as a learning dataset, and MEVIEW as a test dataset and the last one considers both SMIC and CasmeII as a learning dataset, and MEVIEW as a test dataset. The accuracy and F-measure are used to validate these scenarios of experiments. We have considered only three facial micro-expressions for each dataset: *"negative"*, *"positive"*, and *"surprise"*. The results obtained are presented in Table 13.

The expected experimental results of the classifiers of the five scenarios may not be satisfactory, since it shows between 51.85% and 73.47% of recognition rate, and between 0.49 and 0.70 of F-measure. The classifier of scenario 2, that has used SMIC as a learning dataset, and CasmeII as a test dataset, records the best result.

Huang et al., [61] applied the cross dataset for facial micro-expression recognition using several approaches, based on handcrafted features from the literature. The accuracy (resp. F-measure) does not exceed 60% (resp. 0.606) when using SMIC as a learning dataset and CasmeII as a test dataset, and 45.12% (resp. 0.358) when using CasmeII as a learning dataset and SMIC as a test dataset. This prove the effectiveness of our approach that can records 73.47% of accuracy and 0.70 of F-measure using SMIC as a learning dataset and CasmeII as a test dataset.

6. Conclusion and perspectives

The literature on micro-expression recognition shows a variety of methods. Some of these are based on deep learning. The rest, that represent the majority, are based on appearance features and/or motion features. One of the biggest drawbacks of the latter methods is the high size of feature space used to recognize micro-expressions. This paper introduced a facial micro-expression recognition approach based on a low-dimensional feature space called "Uniform Local Binary Patterns on an Accordion 2D representation of sub-regions presented by a Pyramid of levels (LBPAccP^{u2})". The proposed space exploits the effectiveness of uniform LBP patterns from an accordion representation of sub-regions at different sizes. A classification algorithm, based on similarity calculation using RF bootstrap aggregation technique, called *RF_prox*, was also proposed to reduce the classification ambiguity of similar micro-expressions. Extensive experimental studies and comparisons were carried out in order to prove the effectiveness of the proposed approach. The findings of our research are quite convincing, and the following conclusions can be drawn:

- The proposed LBPAccP^{u2} feature space can represent effectively a frontal pose-invariant facial micro-expression video. As a matter of fact, using the accordion technique, transforming the 3D frame-representation of a video sequence into a 2D image that keeps the spatial and temporal information, is more efficient to build a micro-

Table 13
Evaluation results of the proposed approach using cross datasets.

Train/ Test	Accuracy(%)	F-measure	σ (%)
CasmeII/ SMIC	68.29	0.65	2.113
SMIC/ CasmeII	73.47	0.70	1.846
CasmeII/ MEVIEW	51.85	0.49	3.683
SMIC/ MEVIEW	66.67	0.65	3.403
SMIC+CasmeII/ MEVIEW	70.37	0.67	3.123

expression classifier than using the original 3D frame-representation. The use of the accordion technique allows decreasing the number of features necessary in order to present the video sequence. In addition, the pyramid representation can give different weights to the selected sub-regions.
- Using a few discriminating features is enough to create a strong micro-expression recognition classifier that outperforms the famous literature approaches that are based on a high dimensional feature space. Indeed, in this paper, RF is used to select the most discriminating features from LBPAccP^{u2}. This selection was conducted to transform the feature space to a low-dimensional space with a number of features varying between 76 and 91 features depending on the used dataset. Experiments have proved that the selection of the most discriminating feature significantly improves the results.
- Using proximity measures calculated from RF can resolve many conflicts of the micro-expression prediction and considerably improve the performance of the standard RF algorithm. The proximity measure between two videos is defined as the percentage of trees, in the forest, that use the same path to predict the class of these videos. Compared to the standard RF algorithm with feature selection, experiments have shown that RF that uses the proposed proximity measure is the best in terms of the accuracy rate and the F-measure.
- The used dataset has an impact on the performance of the proposed approach. This does not preclude the fact that the proposed approach is one of the best micro-expression recognition approaches in the literature. A comparative study of the proposed approach with some of the famous competitive state-of-the-art approaches proved its performance according to accuracy rate and the number of features. This performance can be remarkably shown on the CasmeII dataset in which we reached 80.81% of accuracy rate using only 79 features.

On the basis of the promising findings presented in this paper, work on the remaining points is underway and there are many future issues that may need further improvements:

- As demonstrated in [15,28], the video magnification can be used to improve facial micro-expression recognition. So, we can explore whether adding a video magnification to our solution, as a pre-processing step, can enhance the performance of our approach.
- Evaluating the performance of our proposal to recognize facial macro-expressions and study if it is possible to create a generic classifier that can recognize both facial macro- and micro-expressions.
- This research was concerned with the recognition of facial micro-expressions of some literature datasets. However, the results should also be applicable to such concrete application fields as lies detection, clinical diagnosis, and teaching assistance.
- As the problem of the imbalanced data is often characteristic of the micro-expression datasets, we are planning to explore if the use of an asymmetric decision tree like AECID_DT [57], in the construction of the forest can be beneficial to improve the performance of the proposed classification algorithm.

CRediT authorship contribution statement

Radhouane Guermazi: Conceptualization, Methodology, Validation, Formal analysis, Writing – original draft. **Taoufik Ben Abdallah:** Conceptualization, Methodology, Software, Visualization, Writing – review & editing. **Mohamed Hammami:** Supervision, Project administration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

xxx

xxx

References

[1] P. Ekman, W.V. Friesen, M. O'Sullivan, A. Chan, Universals and cultural differences in the judgments of facial expressions of emotion, Personal. Soc. Psychol. 53 (1987) 712–717.

[2] M. Takalkar, M. Xu, Q. Wu, Z. Chaczko, A survey: Facial micro-expression recognition, Multimedia Tools Appl. 77 (2018) 19301–19325.

[3] M. O'Sullivan, M.G. Frank, C.M. Hurley, J. Tiwana, Police lie detection accuracy: the effect of lie scenario, Law Hum. Behav. 33 (2009) 530–538.

[4] Imotions, Facial expression analysis: the complete pocket guide, https://imotions.com/blog/Facial-Expression-Analysis, 2016.

[5] M.G. Frank, C.J. Maccario, V. Govindaraju, Behavior and security, in: Book: Protecting Airline Passengers in the Age of Terrorism, Santa Barbara, Calif, 2009, pp. 86–106.

[6] T. Russell, E. Chu, M.L. Phillips, A pilot study to investigate the effectiveness of emotion recognition remediation in schizophrenia using the micro-expression training tool, Clin. Psychol. 45 (2006) 579–583.

[7] L.D. Pool, P. Qualter, Improving emotional intelligence and emotional self-efficacy through a teaching intervention for university students, Learn. Individual Differences 22 (2012) 306–312.

[8] J. Whitehill, Z. Serpell, Y. Lin, A. Foster, J.R. Movellan, The Faces of Engagement: Automatic Recognition of Student Engagement from Facial Expressions, IEEE Trans. Affective Comput. 5 (2014) 86–98.

[9] P. Zhang, X. Ben, R. Yan, C. Wu, C. Guo, Micro-expression recognition system, Light Electr. Opt. 127 (2016) 1395–1400.

[10] H. Zheng, Micro-expression recognition based on 2D Gabor Filter and Sparse Representation, J. Phys. 787 (2017) 1–6.

[11] S.J. Wang, H.L. Chen, W.J. Yan, Y.H. Chen, X. Fu, Face recognition and micro-expression recognition based on discriminant tensor subspace analysis plus extreme learning machine, Neural Process. Lett. 39 (2014) 25–43.

[12] S.J. Wang, W.J. Yan, X. Li, G. Zhao, X. Fu, Micro-expression recognition using dynamic textures on tensor independent color space, in: International Conference on Pattern Recognition, IEEE, Stockholm, Sweden, 2014b, pp. 4678–4683.

[13] W.J. Yan, X. Li, S.J. Wang, G. Zhao, Y.J. Liu, Y.H. Chen, X. Fu, Casmeii: an improved spontaneous micro-expression database and the baseline evaluation, Plos One 9 (2014) 1–8.

[14] Y. Wang, J. See, R.W. Phan, Y.H. Oh, LBP with six intersection points: Reducing redundant information in LBP-TOP for micro-expression recognition, in: Asian Conference on Computer Vision, Springer, Singapore, 2015, pp. 525–537.

[15] Y. Wang, J. See, Y.H. Oh, R.C.W. Phan, Y. Rahulamathavan, H.C. Ling, S.W. Tan, X. Li, Effective recognition of facial micro-expressions with video motion magnification, Multimedia Tools Appl. 76 (2017) 21665–21690.

[16] F. Qu, S. Wang, W. Yan, H. Li, S. Wu, X. Fu, Cas(me)^2: a Database for Spontaneous Macro-Expression and Micro-Expression Spotting and Recognition, IEEE Trans. Affective Comput. 9 (2018) 424–436.

[17] Y. Guo, C. Xue, Y. Wang, M. Yu, Micro-expression recognition based on CBP-TOP feature with ELM, Light Electr. Opt. 126 (2015) 4446–4451.

[18] X. Huang, G. Zhao, X. Hong, W. Zheng, M. Pietikäinen, Spontaneous facial micro-expression analysis using spatiotemporal completed local quantized patterns, Neurocomputing 175 (2016) 564–578.

[19] Y. Zong, X. Huang, W. Zheng, Z. Cui, G. Zhao, Learning from hierarchical spatiotemporal descriptors for micro-expression recognition, IEEE Trans. Multimedia 20 (2018) 3160–3172.

[20] S.T. Liong, J. See, R.C. Phan, K. Wong, S.W. Tan, Hybrid facial regions extraction for micro-expression recognition system, Signal Process. Syst. 90 (2018) 601–617.

[21] X. Huang, S.J. Wang, G. Zhao, M. Piteikainen, Facial micro-expression recognition using Spatiotemporal Local Binary Pattern with Integral Projection, in: International Conference on Computer Vision Workshop, IEEE, Washington, DC, USA, 2015, pp. 1–9.

[22] X. Jia, X. Ben, H. Yuan, K. Kpalma, W. Meng, Macro-to-micro transformation model for micro-expression recognition, Comput. Sci. 25 (2018) 289–297.

[23] X. Ben, X. Jia, R. Yan, X. Zhang, W. Meng, Learning effective binary descriptors for micro-expression recognition transferred by macro-information, Pattern Recogn. Lett. 107 (2018) 50–58.

[24] Y.J. Liu, J.K. Zhang, W.J. Yan, S.J. Wang, G. Zhao, X. Fu, A Main Directional Mean Optical Flow feature for Spontaneous micro-expression recognition, IEEE Trans. Affective Comput. 7 (2015) 299–310.

[25] F. Xu, J. Zhang, J.Z. Wang, Microexpression identification and categorization using a Facial Dynamics Map, IEEE Trans. Affective Comput. 8 (2017) 254–267.

[26] H. Lu, K. Kpalma, J. Ronsin, Motion descriptors for micro-expression recognition, Signal Process. Image Commun. 67 (2018) 108–117.

[27] S.T. Liong, J. See, R.C. Phan, K. Wong, Less Is More: Micro-Expression Recognition from Video using Apex Frame, Signal Process. Image Commun. 62 (2018) 82–92.

[28] X. Li, X. Hong, A. Moilanen, X. Huang, T. Pfister, G. Zhao, M. Pietikäinen, Towards Reading Hidden Emotions: a Comparative Study of Spontaneous Micro-Expression Spotting and Recognition Methods, IEEE Trans. Affective Comput. 9 (2018) 563–577.

[29] L. Wang, H. Xiao, S. Luo, J. Zhang, X. Liu, A weighted feature extraction method based on temporal accumulation of optical flow for micro-expression recognition, Signal Process. Image Commun. 78 (2019) 246–253.

[30] K.M. Goh, C.H. Ng, L.L. Lim, U.U. Sheikh, Micro-Expression Recognition: an Updated Review of Current Trends, Challenges and Solutions, The Visual Computer, 2018, pp. 1–24.

[31] D.H. Kim, W.J. Baddar, Y.M. Ro, Micro-expression recognition with Expression-State Constrained Spatio-Temporal feature representations, in: International Conference on Multimedia, ACM, New York, NY, USA, 2016, pp. 382–386.

[32] M. Peng, C. Wang, T. Chen, G. Liu, X. Fu, Dual Temporal Scale Convolutional Neural Network for Micro-Expression Recognition, Front. Psychol. 8 (2017).

[33] S.J. Wang, B.J. Li, Y.J. Liu, W.J. Yan, X. Ou, X. Huang, F.X., X. Fu, Micro-Expression Recognition with Small Sample Size by Transferring Long-Term Convolutional Neural Network, Neurocomputing 312 (2018) 251–262.

[34] S.P. Reddy, S.T. Karri, S.R. Dubey, S. Mukherjee, Spontaneous facial micro-expression recognition using 3d spatiotemporal convolutional neural networks, in: International Joint Conference on Neural Network, Budapest, 2019, pp. 1–8.

[35] M.A. Takalkar, M. Xu, Z. Chaczko, Manifold feature integration for micro-expression recognition, Multimedia Syst. (2020) 1–17.

[36] Y. Huang, Y. Yan, S. Chen, H. Wang, Expression-targeted feature learning for effective facial expression recognition, Visual Commun. Image Represent. 55 (2018) 677–687.

[37] Z. Yu, G. Liu, Q. Liu, J. Deng, Spatio-Temporal Convolutional Features with Nested LSTM for Facial Expression Recognition, Neurocomputing 317 (2018) 50–57.

[38] R. Zhi, H. Xu, M. Wan, T. Li, Combining 3d convolutional neural networks with transfer learning by supervised pre-training for facial micro-expression recognition, IEICE Trans. Inform. Syst. 102 (2019) 1054–1064.

[39] C. Wang, M. Peng, T. Bi, T. Chen, Micro-attention for micro-expression recognition, Neurocomputing 410 (2020) 354–362.

[40] S. Wang, W. Yan, X. Li, G. Zhao, C. Zhou, X. Fu, M. Yang, J. Tao, Micro-expression recognition using color spaces, IEEE Trans. Image Process. 24 (2015) 6034–6047.

[41] P. Ekman, W.V. Friesen, Facial Action Coding System: a technique for the measurement of facial movement, Consulting Psychologists Press, 1978.

[42] T.B. Abdallah, R. Guermazi, M. Hammami, Towards micro-expression recognition through Pyramid of uniform Temporal Local Binary Pattern features, in: International Conference on Intelligent Systems Design and Applications, Springer, Vellore, India, 2019, pp. 629–640.

[43] T.B. Abdallah, R. Guermazi, M. Hammami, Facial-expression recognition based on a low-dimensional temporal feature space, Multimedia Tools Appl. 77 (2018) 19455–19479.

[44] L. Breiman, Random Forests, Mach. Learn. 45 (2001) 5–32.

[45] T. Ojala, M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based on feature distributions, Pattern Recogn. 29 (1996) 51–59.

[46] T. Ouni, W. Ayedi, M. Abid, New low complexity dct based video compression method, in: International Conference on Telecommunications, IEEE, Marrakech, Morocco, 2009, pp. 202–207.

[47] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 971–987.

[48] B.N. Datta, A Review of Some Basic Concepts and Results from Theoretical Linear Algebra, in: Book: Numerical Methods for Linear Control Systems, chapter2, Academic Press, San Diego, 2004, pp. 19–32.

[49] X. Li, T. Pfister, X. Huang, G. Zhao, M. Pietikäinen, A spontaneous micro-expression database: Inducement, collection and baseline, in: International Conference and Workshops on Automatic Face and Gesture Recognition, IEEE, Shanghai, China, 2013, pp. 1–6.

[50] W.J. Yan, Q. Wu, Y.J. Liu, S.J. Wang, X. Fu, Casme database: a dataset of spontaneous micro-expressions collected from neutralized faces, in: International Conference and Workshops on Automatic Face and Gesture Recognition, IEEE, Shanghai, China, 2013, pp. 1–7.

[51] P. Husák, J. Cech, J. Matas, Spotting facial micro-expressions in the wild, in: 22nd Computer Vision Winter Workshop (Retz), 2017, pp. 1–9.

[52] T.F. Cootes, C.J. Taylor, D.H. Cooper, J. Graham, Active Shape Models: their Training and Application, Comput. Vis. Image Underst. 61 (1995) 38–59.

[53] Z. Niu, S. Shan, S. Yan, X. Chen, W. Gao, 2d cascaded adaboost for eye localization, in: International Conference on Pattern Recognition (ICPR'06), IEEE, Hong Kong, China, 2006, pp. 1216–1219.

[54] A. Asthana, S. Zafeiriou, S. Cheng, M. Pantic, Robust discriminative response map fitting with Constrained Local Models, in: Conference on Computer Vision and Pattern Recognition, IEEE, Portland, OR, USA, 2013, pp. 3444–3451.

[55] T.M. Oshiro, P.S. Perez, J.A. Baranauskas, How many trees in a random forest?, in: International Workshop on Machine Learning and Data Mining in Pattern Recognition Springer, Berlin, Heidelberg, 2012, pp. 154–168.

[56] I. Chaabane, R. Guermazi, M. Hammami, Enhancing techniques for learning decision trees from imbalanced data, Adv. Data Anal. Classif. (2019) (To appear).

[57] R. Guermazi, I. Chaabane, M. Hammami, AECID: Asymmetric Entropy for Classifying Imbalanced Data, Inf. Sci. 467 (2018) 373–397.

[58] J.L. Hodges, E.L. Lehmann, Ranks methods for combination of independent experiments in analysis of variance, Ann. Math. Stat. 33 (1962) 482–497.

[59] I. Rodríguez-Fdez, A. Canosa, M. Mucientes, A. Bugarín, STAC: a web platform for the comparison of algorithms using statistical tests, in: International Conference on Fuzzy Systems, IEEE, Istanbul, Turkey, 2015, pp. 1–8.

[60] S. García, A. Fernández, J. Luengo, F. Herrera, Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power, Inf. Sci. 180 (2010) 2044–2064.

[61] T. Zhang, Y. Zong, W. Zheng, C.P. Chen, X. Hong, C. Tang, Z. Cui, G. Zhao, Cross-database micro-expression recognition: A benchmark, IEEE Trans. Knowl. Data Eng. (2020).