Time Unification on Local Binary Patterns Three Orthogonal Planes for Facial Expression Recognition

Reda Belaiche, Cyrille Migniot, Dominique Ginhac, Fan Yang

Time Unification on Local Binary Patterns Three Orthogonal Planes for Facial Expression Recognition

Reda Belaiche, Cyrille Migniot, Dominique Ginhac & Fan Yang
ImViA EA 7535, Univ. Bourgogne Franche-Comté, Dijon, France
{Name.Surname}@u-bourgogne.fr

***Abstract*—Machine learning has known a tremendous growth within the last years, and lately, thanks to that, some computer vision algorithms started to access what is difficult or even impossible to perceive by the human eye. While deep learning based computer vision algorithms have made themselves more and more present in the recent years, more classical feature extraction methods, such as the ones based on Local Binary Patterns (LBP), still present a non negligible interest, especially when dealing with small datasets. Furthermore, this operator has proven to be quite useful for facial emotions and human gestures recognition in general. Micro-Expression (ME) classification is among the applications of computer vision that heavily relied on hand crafted features in the past years. LBP Three Orthogonal Planes (LBP_TOP) is one of the most used hand crafted features extractor in the scientific literature to tackle the problem of ME classification. In this paper we present a time unification method that provides better results than the classical LBP_TOP while also drastically reducing the calculations required for feature extraction.**

I. Introduction

Facial expressions offer important benchmarks in every day's social interactions. Most people are familiar with macro facial expressions. However few people are aware of the existence of micro-facial expressions [1] [2], and even fewer know how to detect and recognize said Micro-Expressions. Initially discovered by Haggard and Isaacs [3], Micro-Expressions are a type of involuntary facial expressions that are extremely fast and of very low intensity. Their duration is very short, which makes their localization and analysis rather complicated tasks. Micro-Expressions (ME) occur in two situations: conscious suppression and unconscious repression. Conscious suppression happens when a person intentionally tries to stop themselves from showing their true emotions or try to hide them. Unconscious repression occurs when the subject himself does not realize their true emotions. In both cases, micro-expressions betray the subject's real emotions independently from his awareness of their existence. Ekman also proved that Micro-Expressions are not culturally dependent but universal [2] which makes developing automatic ways to classify them even more interesting.

In sum, Micro-Expressions are involuntary, universal and expose a persons true emotions. They are characterised by their short duration (0.04s-0.25s *vs* 0.5s-4s for Macro-expressions) and low intensity. Facial expressions in general begin at the onset frame, reach their maximum intensity at the apex frame, and finish at the offset frame (Fig. 1).

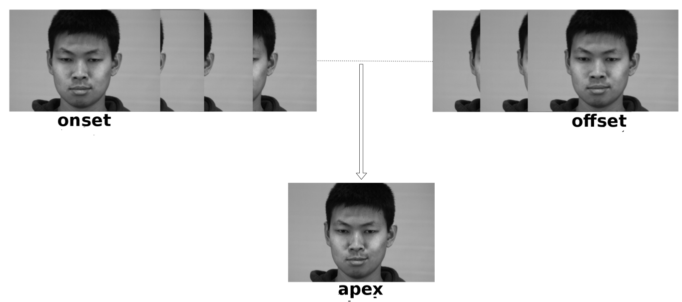

Fig. 1. Example of a Micro-Expression: the maximum intensity occurs at the apex frame.

Automatic ME recognition can be used in many real world applications such as neuromarketing [4] or automobile drivers' monitoring [5].

Local Binary Patterns [6] based operators have been extensively used in the last decade. Its three dimensional variant, *Local Binary Patterns Three Orthogonal Planes* (LBP_TOP), has been thought of as a solution in many studies working on classifying three dimensional data : be it three dimensional images or two dimensional videos, helping to classify different kinds of videos and data [7] [8] [9] [10] [11] [12].

The LBP_TOP operator has proven it's usefulness in many studies, but when used as a spatio-temporal descriptor for two dimensional videos, it can prove to be quite expensive in terms of computations. The principal cause being that the descriptor considers a series of frames as a three-dimensional matrix and go over all the frames present in it, applying the LBP operator on each pixel with it's three dimensional neighborhood. We believe that scanning all the frames might not be useful as there might be very little movement between some subsequent frames, thus we advocate using sub-sampling on too long frame series. We also notice that comparing frames series that have the same number of frames might be more pertinent than comparing unequal frame series. So we add a time interpolation model to frame series that are too small. Time unification proposed in this paper consists in temporally resizing all the frame series to the same fixed number of frames. Not only does it make the classification of the features extracted easier, but unifying the frame number to a small number (thus making features extraction much less expensive in terms of computations) provides the best results.

In this paper, we propose to use a time unification step before using the LBP_TOP operator as feature extractor and test it on Micro and Macro-Facial Expressions(M/M-FEs) classification with the $CAS(ME)^2$ dataset [13].

The paper is organized as follows: we describe M/M-FEs and PRV-based feature extraction method used for this study, namely the LBP_TOP operator, in section II. Experiments are presented and discussed in section III and a conclusion is given in section IV.

II. Feature extraction method

A. LBP_TOP

1) LBP: In order to recognize Micro and Macro Expressions, we have to go through two steps: feature extraction and classification. M/M-FEs can be described by a spatio-temporal local texture descriptors. LBP operator is a visual descriptor that was originally designed for texture description [6]. Ahonen et al [14] used it for face recognition in view of its efficiency and fast feature extraction. They claimed superiority over all the other algorithms they compared in tests on the FERET database. The general idea is to threshold a small area around each pixel in order to build a binary code. This code is obtained by comparing neighbour pixel values with the center pixel: values superior or equal to the center pixel's value get assigned a 1 while smaller values get assigned a 0. The choice of the surrounding area directly affects the kind of edges it is possible to detect in an image. For a neighborhood referring to *N* sampling points on a circle of radius *R*, the LBP binary value of a pixel c is given by:

$$LBP_{N,R} = \sum_{p=0}^{N-1} t(g_p - g_c)2^p \tag{1}$$

Here g_c represents the gray value of the center pixel c while g_p represents the gray value of a pixels in the neighborhood N, R. t defines a thresholding function: $t(x) = 1$ if $x \geq 0$ and $t(x) = 0$ otherwise. The feature vector representing an input image is calculated by extracting the histogram distribution of the LBP on all the pixels.

We can consider LBP as texture primitives that include different types of curved edges, spots, flat areas, and so on. For an efficient facial representation, images usually get divided into local blocks from which LBP histograms are extracted. Local texture can be described using said histograms that are then concatenated into an enhanced feature histogram [14]. The number of blocks and the size of each block determines the level of retained spatial information.

2) Extention to LBP_TOP: The conventional LBP only serves for spatial data in 2D images. To describe data from the 3D spatio-temporal domain, a frame series is considered as a 3D object and the basic LBP is extracted following the three directions given by the planes XY, XT and YT for each pixel as shown in Fig.2. The resulting three histograms are then concatenated. After being originally proposed for dynamic textures description [15], it was first used for Micro-Expressions recognition by Pfister *et al.* [16] .

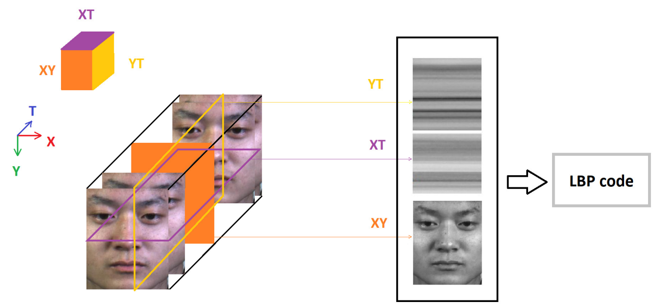

Fig. 2. Illustration of a spatio-temporal volume of a video [13]

B. Time unification

Unlike spacial components, the temporal component of the frame series varies with the expressions. Indeed, different facial expressions can have different duration. Not having the constrain of time unity seem to confer an advantage, but it also means that it's possible to compare matrices with a very different time component. For example, having to classify the histograms extracted from a series of 60 and 8 frames using the exact same method might not be optimal. We propose to mix sub-sampling and linear interpolation to have the same number of frame over all the facial expressions present in the dataset.
Sub-sampling gets rid of superfluous information by not incorporating too closed images. It reduces the calculation required to extract features. Time interpolation increases too small samples.
Our goal is temporally resizing frame series so as to make all of them have the same number of frames (we named $desired\_size$). According to the initial number of frame of the frame series (named $original\_size$), sub-sampling is processed if $original\_size > desired\_size$ and time interpolation if $original\_size < desired\_size$.

1) Sub-sampling: The information on the action over all the frame series must be kept intact, especially around the occurrence of the apex frame that concentrates most of the movement. We first determine an interval ($jump$) between two successive frames that will be preserved. $jump$ is related to the quotient of the original size of the frame series and by the desired size. If $desired\_size$ does not divide the $original\_size$, we complete the new sample by frames close to the apex. Algorithm 1 is used for sub-sampling.

2) Time interpolation: The value of the variable $jump$ is calculated similarly to with sub-sampling. After the initial distribution is done, we keep adding linear interpolations between each subsequent frames until we reach the desire number of frames for our new frame series.

Data: Fame series $frame\_series$, number of frames $original\_size$, desired number of frames $desired\_size$, index of the apex id_apex
Result: Sub-sambled frame series new_frame_series with $desired\_size$ frames

```
jump=ceil(original_size/desired_size);
i=1;
j=1;
while i < desired_size and j < original_size do
    new_frame_series[i]=frame_series[j];
    i=i+1;
    j=j+jump;
end
if i = 0 then
    return new_frame_series;
else
    while i > 0 do
        insert un-inserted frames around id_apex from
          frame_series to new_frame_series;
    end
    return new_frame_series;
end
```

Algorithm 1: Time sub-sampling algorithm

III. Experiments and discussion

A. Dataset presentation

The number of scientific papers dealing with the automatic analysis of Micro-Expressions is rather limited. One of the reasons for it can be attributed to the lack of datasets containing real Micro-Expressions. Fortunately, this is beginning to change and new foundations are being laid, with new datasets relating spontaneous Micro-Expressions. As a matter of fact, the dataset we work on, $CAS(ME)^2$ dataset [13], is one of the few datasets to present annotated videos of spontaneous M/M-FEs of different subjects. 22 participants in total were filmed while watching different kinds of excitation videos. Each subject was informed that their monetary rewards would be depend on their ability to suppress their facial expressions, thus promoting the appearance of Micro-Expressions while also reducing the intensity of macro expressions present in the dataset.

This dataset was originally proposed for automatic M/M-FEs spotting and recognition, and while Micro-Expressions spotting has been getting good results [17], their recognition still offers many challenges.

$CAS(ME)^2$ offers annotations for M/M-E that are based on the facial muscle movements according to the *Action Units* (AU) following the Facial Action Coding System (FACS) proposed by Ekman. The position of the apex frame is also provided in this dataset. The low intensity of the Micro Expressions is one of the biggest challenges for their classification.

B. Model validation protocol

Classification was tested following the *Leave One Subject Out* (LOSO) cross-validation protocol: one subject's data is used as a test set in each fold of the cross-validation. This is done to better reproduce actual use conditions where the encountered subjects are alien to the model when it was trained. Using k-fold cross-validation and randomly distributing the different samples in the dataset might result in severe case of overfitting as the accuracies on the training sets would be much higher than with LOSO. This can be attributed to the fact that samples from the same subject would be present in both the training and testing sets. Considering the fact that the same subject can show the same expression many times (which may cause that expression to belong to the training and the test sets at the same time), and that some subjects can be more inclined to show a specific type of emotion more often, using the LOSO protocol seems to be the most rigorous option.

C. Results

The feature vector obtained by LBP_TOP is given as input to an SVM with an RBF kernel with Hyperparameter Optimization Options set to $expected-improvement-plus$ for classification.

We tested time unification with different values of $desired\_size$ and computed the model's accuracy for each of them. The results are presented in Fig. 3 :

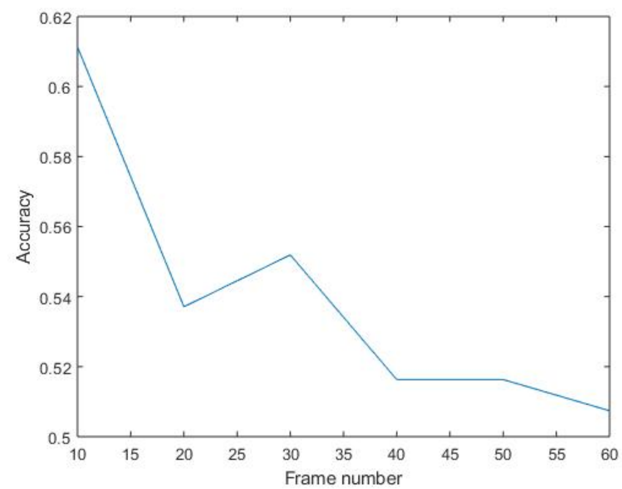

Fig. 3. Overall accuracy (in %) of the model given by different frame unification numbers

We observe that the best score is achieved by the smallest frame unification number. The bigger is the frame unification number, the lower the model's accuracy gets. A model with small frame numbers is also faster and still more accurate.

Fig.4 shows the confusion matrix corresponding achieved by a model with no frame unification and Fig. 5 shows the confusion matrix corresponding to the best accuracy for a model with frame unification number. The model performs better with time unification in all the classes with no exception. We can see that model is not competent for recognising surprise. The reason for that is that only 6.74% of the dataset shows surprise M/M-FEs while 32.55% of the dataset amounts to positive, 36.66 is negative and 24.05% is accounted as other.

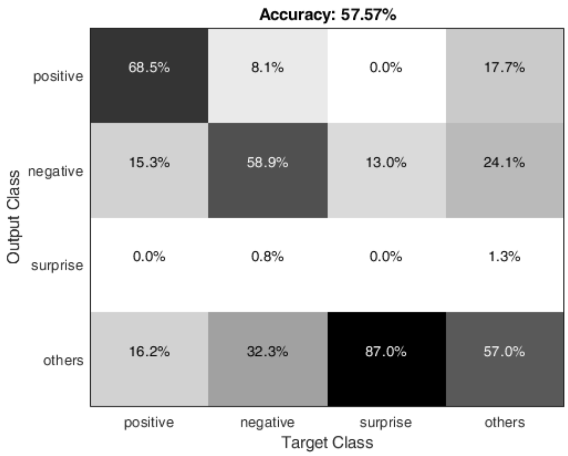

Fig. 4. Confusion matrix of a model without time unification.

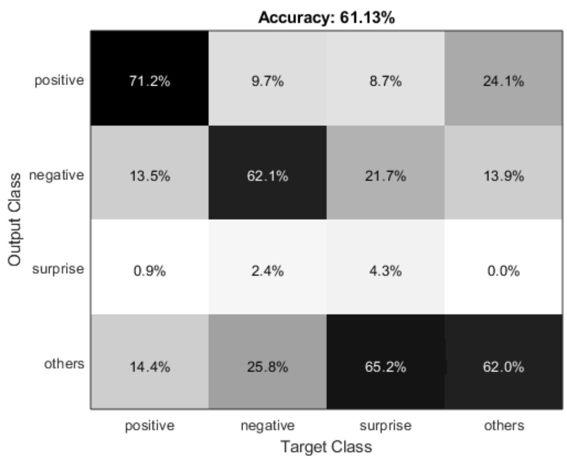

Fig. 5. Confusion matrix of a model with time unification ($desired\_size = 10$).

When comparing the two scores, we observe that the model with frame unification number set at 10 performs better than the one with no frame unification. Not only is the model slightly more accurate, but the feature extraction for the model with frame unification is much less expensive than the model without it : We used a mid-range computer with an Intel i5 8th generation processor and no graphic card to do the calculations. Matlab 2018a was used to write the code. We found that, for feature extraction, only 0.19 seconds are necessary when the number of frames equals 10 while 4.78 seconds are needed when the number of frames equals 70.

IV. Conclusion

LBP-TOP has shown promising performances on facial expression recognition as well as other tasks such as sign language and human activity analysis. As it extracts features from spatio-temporal information, the calculations for LBP-TOP have to go over all the pixels in the three dimensional space, which is computationally expensive.

We have showed that for M/ME recognition, it is possible to extract features from frame series with a unified frame number. This process not only makes the features extracted more appropriate, but also makes feature extraction faster. The required calculations are reduced while still keeping a good accuracy.

References

[1] P. Ekman and W. V. Friesen, “Nonverbal leakage and clues to deception,” *Psychiatry*, 1969.

[2] P. Ekman, *Telling Lies: Clues to Deceit in the Marketplace, Politics, and Marriage (Revised Edition)*, WW Norton & Company, 2009.

[3] E.A. Haggard and K.S. Isaacs, *Methods of Research in Psychotherapy*, Springer, 1966.

[4] G. Vecchiato, L. Astolfi, and F. D. V. Fallani, “On the use of eeg or meg brain imaging tools in neuromarketing research,” *Computational Intelligence and Neuroscience*, 2011.

[5] C. Nass, M. Jonsson, and H. Harris, “Improving automotive safety by pairing driver emotion and car voice emotion,” *Extended Abstracts on Human Factors in Computing Systems*, 2005.

[6] T. Ojala, M. Pietikinenand, and T. Menp, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2002.

[7] M. S. Anju Panicker et al, “Cardio-pulmonary resuscitation (cpr) scene retrieval from medical simulation videos using local binary patterns over three orthogonal planes,” *International Conference on Content-Based Multimedia Indexing*, 2018.

[8] X. Hong, Y. Xu, and G. Zhao, “Lbp-top: A tensor unfolding revisit,” *Asian Conference on Computer Vision*, 2017.

[9] D. Feng and F. Ren, “Dynamic facial expression recognition based on two-stream-cnn with lbp-top,” *IEEE International Conference on Cloud Computing and Intelligence Systems(CCIS)*, 2018.

[10] G. Piantadosi, R. Fusco, and A. Petrillo, ,” .

[11] Y. Wang, H. Yu, B. Stevens, and H. Liu, “Dynamic facial expression recognition using local patch and lbp-top,” *8th International Conference on Human System Interactions (HSI)*, 2015.

[12] A. Saleh and M. Safaa, “Arabic sign language recognition using spatio-temporal local binary patterns and support vector machine,” *Advanced Machine Learning Technologies and Applications pp 36-45*, 2014.

[13] F. Qu, S.J. Wang, and W.J. Yan et al., “Cas(me)^2: A database for spontaneous macro-expression and micro-expression spotting and recognition,” *IEEE Trans. on Affective Computing*, 17 January 2017.

[14] T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2006.

[15] G. Zhao and M. Pietikainen, “Dynamic texture recognition using local binary patterns with an application to facial expressions,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2007.

[16] T. Pfister, X. Li, G. Zhao, and M. Pietikainen, “Recognising spontaneous facial micro-expressions,” *IEEE Inter. Conf. on Computer Vision*, 2011.

[17] Y. Han, B. Li, Y. K. Lai, and Y. J. Liu, “Cfd: A collaborative feature difference method for spontaneous micro-expression spotting,” *25th IEEE ICIP*, 2018.