Encoding Local Binary Patterns Using the Re-Parametrization of the Second Order Gaussian Jet

John A. Ruiz-Hernandez and Matti Pietikäinen

Abstract—In object recognition a robust feature set is considered as an important component in almost all the approaches proposed in the literature. In facial analysis, one of the best known feature set is based in Local Binary Patterns (LBP) which extracts the information contained in the image using comparisons between pixels in a region, finally such comparisons are encoded in form of histogram. We argue that this kind of encoding is statistically non-stable and can lead to errors during the recognition process, specially in noisy and low-resolution images, where the information contained in the image is not enough to generate a statistically robust histogram. In this paper, we propose a new method to encode the Local Binary Patterns using an re-parametrization of the second local order Gaussian Jet which generates more robust and reliable histograms suitable for different facial analysis tasks. We show that our method can be used for recognizing micro-expressions with competitive performances on the Spontaneous Micro-expression Corpus (SMIC) and the YORK Deception Detection Test.

I. INTRODUCTION

Despite the highly developed ability of humans to obtain information from visual observation of faces, facial analysis remains a very challenging task for computer vision. This challenge is specially noticeable when rapid involuntary facial expressions or micro-expressions have to be recognized. Over the last few decades, researchers in computer vision have explored a variety of approaches to obtain information from facial images. While most approaches have proven highly sensitive to image noise, illumination and view position, the occasional successes have often signalled the emergence of important new paradigms for computer vision. The methods that currently dominate the field are based on the use of high-dimensional feature sets that use machine learning discriminants such a kernel methods.

In the task of facial analysis, both the feature set and the machine learning method are crucial for obtaining an optimal performance. Here we are concentrated on the feature set. We introduce a new quantization method based on an angular re-parametrization of the second local order Gaussian Jet. In the proposed approach, the angular re-parametrization is used to improve the performance of the final feature vector. This improvement is due to the discriminative properties of the Gaussian derivatives and the transformation of the feature vector in an angular model which enhances the invariance to some spatial transformations and lighting problems which can be present on images taken in real-world conditions, specially, on those related with facial analysis.

Finally as additional property, the proposed approach is suitable for static or dynamic visual features. Particularly, we use a quantified version of LBP-TOP as low-level feature to extract textural information from XY-T volumes of faces and recognizing facial micro-expressions.

To explain the concepts in our approach, the paper has been organized as follows. In Section II, we present some recent works that inspire the proposed approach. In Section III, we explain our proposed method for encoding using Gaussian derivatives of second order, and, Section IV provides the results of applying the proposed method for recognizing micro-expressions. Finally we summarize our findings in Section V.

II. PREVIOUS WORK

In object recognition, numerous low-level features have been proposed as a solution to extract robust information from facial images. On the scope of this paper, we are interested in two kinds of approaches: 1) the features based on local patterns of qualitative gray-level differences including Local Binary Patterns [1] and its extension to spatio-temporal structures LBP-TOP [2]. 2) features based on oriented histograms.

Considering features based on encoding local patterns, some of the most representative examples are Haar-like features [3], Local Ternary Patterns (LTP) [4], Bio-Inspired features [5], Region Covariance [6], Tensorial Representations of Gaussian Feature Maps [7], [8], Discriminant Image filters with LBP [9] and finally, Spatiotemporal Local Monogenic Binary Patterns (STLMBP) [10] which is an advanced extension of LBP-TOP.

The features based on local patterns are suitable for tasks that require high computational speed and perform well in facial analysis tasks when combined with a robust machine learning method. Nevertheless, in some cases they are not robust to spatial transformations and changes in illumination due to their simplicity which could limit their use in complicated facial analysis tasks. Besides, the encoding used in LBP could generate statistically non-reliable histograms [11].

Regarding histogram feature sets based on angular information, specially on oriented image gradients, some of the most popular approaches owing to their performance and efficiency are SIFT [12], SURF [13], DAISY [14], Histogram of Oriented Gradients (HOG) [15], PHOG [16], Generalized Shape Context [17], Histograms of Receptive Fields [18] and Local Edge Orientation Histograms [19]. In addition, other related approaches include: STIP [20], Dense Trajectory based HOG(DTF-HOG) [21] and Dense Trajectory HBM.
(DTF-MBH) [22] which are extensions to spatio-temporal frameworks of the histogram based on image gradients. The advantage of these methods is their highly robustness to spatial transformation on the images, which make them suitable for many tasks in facial analysis and 3D reconstruction.

Our method differs from the aforementioned in the following aspects:

- LBP-TOP is encoded using angular representations rather than simple histograms, this improvement enhances the statistical stability and improves the discriminative power in the computed histograms.
- We encode using a set of angular representations based in the orientation of the gradient and the Gaussian derivatives of second order which can describe more complicated shapes such as bars, blobs and corners, as consequence the captured information can improve considerably the performance in an object recognition system.
- In our approach, the information provided by the encoding at different sigma values ($\sigma$) is considered separately. The encoding using lower values of sigma provides higher details from the image but is sensible to noise, in contrast higher values of sigma provide less details but could be useful for dealing with problems related with noise. In addition, we consider a machine learning approach to combine the information provided for each value of sigma.

III. OUR APPROACH

The proposed approach is summarized in Figure 1. The two main blocks are “Re-parametrization” and “Encoding. In this section, we are going to explain in details each one of these operations from mathematical and practical perspectives.

A. Re-parametrization of the second order Gaussian jet

From a mathematical point of view, computing the Gaussian derivatives on an image $I(x,y)$ can be defined by the equation:

$$I_{m,n}(x,y,\sigma) = (\frac{\partial^{m+n}G(x,y,\sigma)}{\partial x^{m}\partial y^{n}}) * I(x,y)$$ (1)

where $(m,n)$ are the order of the derivatives in the canonical directions $(x,y)$ respectively, $*$ corresponds to the convolution operation and $G(x,y,\sigma) = (1/\sqrt{2\pi}) * e^{-((x^2+y^2)/2\sigma^2)}$ corresponds to the Gaussian support with scale value $\sigma$.

Koenderink and van Doorn[23] argue that the local visual appearance in an image neighbourhood can be represented by a local Taylor series expansion of the neighbourhood, computed using local Gaussian derivatives. The coefficients of this series constitute a feature vector, referred to as the “Local Gaussian Jet” $\vec{j} = (I_x, I_y, I_{x^2}, I_{x^2}, I_{xy})$ that compactly represents image appearance and can be used for indexing, matching and recognition. Finally to complete the definition of the space formed by the local jet, Griffin [24] defined its magnitude $\|\vec{j}\|$ as:

$$\|\vec{j}\| = \left((\sigma^2 I_x^2 + I_y^2) + 1/4\sigma^4 I_x^2 + I_y^2 + 1/4\sigma^4 (I_{x^2} - I_{xy})^2 + 4I_{xy}^2\right)^{1/2}$$ (2)

Using the advantages offered by the Gaussian jet for recognizing objects and the mathematical properties of its norm, Griffin [24] proposes a new re-parametrization (see eq. (3)) of the vector $\{I_x, I_y, I_{x^2}, I_{xy}\}$ into a new space of coordinates where most of the parameters are not affected by different transformations such as rotation, translation, reflection, affine scaling and intensity.

$$\theta = \tan^{-1}\left(\frac{I_y}{I_x}\right)$$
$$l = \tan^{-1}\left(\sigma(I_{x^2} + I_{xy})\sqrt{4(I_x + I_y)^2 + \sigma^2 (I_{x^2} - I_{xy})^2 + 4I_{xy}^2}\right)^{1/2}$$
$$b = \tan^{-1}\left(\sigma\sqrt{\frac{(I_{x^2} - I_{xy})^2 + 4I_{xy}^2}{4(I_x^2 + I_y^2)}}\right)$$ (3)
$$a = 0.5 \tan^{-1}\left(\frac{2(I_x^2 - I_y^2)I_{xy} + I_x (I_{x^2} - I_{xy})}{(I_x^2 - I_y^2) (I_{x^2} - I_{xy}) + 4I_x I_y I_{xy}}\right)$$
With domains defined as follows $\theta \in (-\pi, \pi]$, $l \in [-\pi/2, \pi/2]$, $b \in [0, \pi/2]$ and $a \in [0, \pi/2]$.

To compute the parametrization angles in a video sequence, the same procedure is applied at every frame in the sequence (see Algorithm 1), the result is a set of 3D blocks defined in terms of $(x, y, t, \sigma, parameter)$ (see Figure 2):

\[
\{\theta(x, y, t, \sigma), l(x, y, t, \sigma), b(x, y, t, \sigma), a(x, y, t, \sigma)\}
\]

(4)

If the speed of the algorithm is important, the values of the Gaussian derivatives can be calculated with a high accuracy and in real time using a Half-Octave Gaussian pyramid [25].

**Algorithm 1:** Re-parametrization in a sequence of video

**Input:** A sequence of video composed by $n$ frames and a set of $p$ sigma values \(\{\sigma_1, \sigma_2 \ldots \sigma_p\}\).

**Result:** A set of 3D blocks defined in terms of the type of parameter and the sigma values \(\sigma\).

1. for $i = 1 : n$ do
2.   for $k = 1 : p$ do
3.     Compute the second local order Gaussian jet \((l_1, l_2, l_3, l_4, l_5)\) with sigma value \(\sigma_k\) on the frame $i$;
4.     Compute the parameters $\theta$, $l$, $b$, and $a$ using the calculated jet from the precedent steep;
5.     Organize the resulting image in the corresponding 3D structure: \(\theta(x, y, t_n, \sigma_p); l(x, y, t_n, \sigma_p); b(x, y, t_n, \sigma_p); a(x, y, t_n, \sigma_p)\)
6.   end
7. end

B. Encoding LBP-TOP using the re-parametrization

For the specific task of image classification, a robust representation of image information is desirable. In this section, we are going to present details of the proposed encoding. In terms of robustness, the proposed encoding takes into consideration the textural spatio-temporal information captured by LBP-TOP and the invariance properties of the angular parameters $\theta$, $l$, $b$ and $a$ which are computed using the second local order Gaussian jet.

For encoding LBP-TOP using the parameters described in Section III-A, an input video sequence is prepared following the next procedure:

1) An LBP-TOP operator \((R = 2\) and bilinear interpolation to find the exact values in the pixel’s neighbours) is applied on an input video sequence, the result is a set of three \((LBP_{XY}(x, y, t), LBP_{XT}(x, t, y)\) and \(LBP_{YT}(y, t, x))\) volumes which are the result of applying LBP on each one of the orthogonal planes of the input video sequence \(((XY), (XT), (YT))\).

2) The procedure explained in Algorithm 1 is applied on the same original video sequence used as input in the precedent steep.

3) Each image corresponding to the third dimension of the resulting LBP blocks is divided into non-overlapping $N$ rectangular sub-regions with an specific size $(d_1 \times d_2)$ (for an example, see Figure 3).

Using the procedure mentioned above, the textural information is captured in a dense space of features by the LBP blocks and the non-overlapping rectangular subregions. Finally this information is encoded with the angular parameters $\theta$, $l$, $b$ and $a$ using Algorithm 2.

**Algorithm 2:** Encoding LBP-TOP using the re-parametrization

**Input:** The set of LBP-TOP blocks prepared using the procedure mentioned in this section, a set of reference vectors (one vector per parameter) denoted as \(\text{ref}\) and the 3D structure of parameters computed using the Algorithm 1

**Result:** Matrix \(\mathcal{M}\) of encoded LBP

1. initializing temporal vector \(\text{TempVec}\) to the adequate size;
2. foreach \(\sigma_p\) value used in the 3D structure of parameters do
3.   foreach \(LBP\) block do
4.     foreach position \((x, y)\) in rectangular subregion $N$ do
5.       weighting the LBP value using
6.       \[
6.       \text{TempVec} = \left(\sum \text{LBP}_{\text{block}}(x, y, t) \cdot \mathcal{N}(\text{ref}, 1)\right)
6.       \]
7. end
8. Normalize \(\text{TempVec}\) to standard deviation 1 and mean 0;
9. \(\mathcal{M}(\sigma,:) = \text{concatenate}(\mathcal{M}(\sigma,:), \text{TempVec});\)
10. \(\text{TempVec} = [\ ]\)
11 end

Basically, the encoding process consists of a weighting operation (see Equation (5) in Algorithm 2) which uses the product of the LBP values with a weighting function to ensure its correct distribution in the feature vector, in our encoding, the inclusion of a normal distribution $\mathcal{N}(\text{ref}, 1)$ guarantees a higher robustness against noise and spatial transformations present on the image, besides the resulting vectors become more statistically reliable compared with the original histograms used in LBP-top. Figure 3 shows an example of our encoding procedure.

IV. MICRO-EXPRESSIONS RECOGNITION EXPERIMENTS

We use the encoding method described in Section III-B as a solution for recognizing micro-expressions. In this section we describe the experimental issues and obtained results.
A. Experimental datasets

Two public available datasets were used in all the experiments described in this paper: the YORK Deception Detection dataset [26] and the Spontaneous Micro-expression Corpus (SMIC) [27].

1) The YORK Deception Detection dataset [26]: the original dataset is composed of 20 videos of resolution 320 × 240 recorded as a part of a psychological study. Using, these 20 videos, Pfister et al [27] segmented the micro-expressions from the original database and labelled them as truthful/deceptive (lie/truth) and emotional/non-emotional (emo/¬ emo). The final database contains 9 subjects (3 male and 6 female) for a total of 18 micro-expressions distributed as follows: 7 emotional and 11 non-emotional; 11 deceptive and 7 truthful. The shortest segmented expression has a duration of 7 frames at the 25fps video frame rate.

2) The Spontaneous Micro-expression Corpus (SMIC): Recorded by Pfister et al [27] to deal with the problems founded in the YORK dataset (small training sample size and low-resolution images). The SMIC dataset consists of 6 subjects (3 male and 3 female) with 77 spontaneous micro-expressions which were recorded in a controlled scenario using 100 fps camera with resolution of 640 × 480. The shortest recorded expression was about 11 frames at 100fps and the average expression length was 29 frames at the same recording speed.

B. Experimental set-up

We evaluate the proposed encoding using a leave-one-subject-out validation in both databases. In all the experiments, for improving the recognition results, the faces were cropped and normalized to a size of 66 × 66 pixels using the positions provided by a Haar-feature-based eye detector and an Active Shape Model (ASM) with 68 feature points, in addition no attempt to correct the problems of lighting in the datasets was performed.

This methodology of normalization follows the experimental protocol used by Pfister et al [27] and allows us to do exact experimental comparisons with his method, which to our knowledge is the first automatic method for recognizing spontaneous micro-expressions.

In the recognition process, we use three machine learning methods (all of them have publicly available Matlab® implementations provided by the authors) based in linear kernels and polynomial kernels of orders 3 and 5 (denoted in the rest of the paper as Linear, poly=3 and poly=5 respectively). The algorithms used to find the classifier were SVM[28], LibLinear [29] and Localized Multiple Kernel Learning [30]. To optimize the common parameter C in all the algorithms, we use a 5-fold cross validation (trying values 0.001, 0.01, 0.1, 1, 10, 100 and 1000) in the training dataset and the best value is used as parameter to train and test the final classifier.

For computing the matrix , we use rectangular sub-regions of size 4 × 4 pixels (3 or 5 pixels from the image’s borders are not considered), sigma values . As reference vectors we use:

\[
\begin{align*}
ref_{\theta,l,b,a} &= \{0 : \pi/9 : \pi\} \\
ref_{\theta,l,b,a} &= \{0 : \pi/18 : \pi/2\}
\end{align*}
\]

Finally, in the recognition process using MKL, we consider kernel feature spaces composed by an specific value of and a specific type of kernel which is reported in each one of the experiments.

C. Experiments using different values with a single kernel

The aim of this experiment is to evaluate the performance of our approach when only a single value from the vector of σ’s is used during the encoding process, besides, only one kernel will be considered without taking into a count a combination using Multiple Kernel Learning.

We report in Table I, the results for this experiments on the YORK dataset. The best results in all the cases were obtained using methods with non computational expensive kernels such as LIBLINEAR or SVM-Linear. In both cases the best value of σ was 0.5√2.

In detection the best recognition rate was 0.724% using an SVM-Linear classifier. In emo/¬ emo classification, the recognition rate was 0.556 % using a classifier trained with LIBLINEAR. Finally, the only exception to the methods using linear kernels was in lie/truth classification where the best result was 0.759 using a SVM-Poly=5 with σ = 1.5√2.

The Recognition Rate scores on the SMIC dataset when the sigma values are used separately for our method are given in Table II. The obtained results were similar to those on the

\footnote{Matlab® implementations of the MKL algorithms are available on-line at http://www.cmpe.boun.edu.tr/~gonen/mkl}
TABLE I: Recognition rate scores of the encoding on the YORK dataset using different values of $\sigma$ and single kernel algorithms

<table>
<thead>
<tr>
<th>$\sigma = 0.5\sqrt{2}$</th>
<th>$\sigma = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIBLINEAR 0.696</td>
<td>LIBLINEAR 0.696</td>
</tr>
<tr>
<td>SVM-Linear 0.724</td>
<td>SVM-Linear 0.696</td>
</tr>
<tr>
<td>SVM-poly=3 0.696</td>
<td>SVM-poly=3 0.683</td>
</tr>
<tr>
<td>SVM-poly=5 0.619</td>
<td>SVM-poly=5 0.702</td>
</tr>
</tbody>
</table>

TABLE II: Recognition rate scores of the encoding on the SMIC dataset using different values of $\sigma$ and single kernel algorithms

<table>
<thead>
<tr>
<th>$\sigma = 0.5\sqrt{2}$</th>
<th>$\sigma = 1.5\sqrt{2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIBLINEAR 0.714</td>
<td>LIBLINEAR 0.665</td>
</tr>
<tr>
<td>SVM-Linear 0.702</td>
<td>SVM-Linear 0.609</td>
</tr>
<tr>
<td>SVM-poly=3 0.637</td>
<td>SVM-poly=3 0.711</td>
</tr>
<tr>
<td>SVM-poly=5 0.637</td>
<td>SVM-poly=5 0.674</td>
</tr>
</tbody>
</table>

TABLE III: Comparisons of Recognition rate scores of the encoding with the state-of-the-art approaches on the YORK dataset

<table>
<thead>
<tr>
<th>Method</th>
<th>detection</th>
<th>emo/~ emo</th>
<th>lie/truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP + SVM(pol=6) [27]</td>
<td>0.650</td>
<td>0.695</td>
<td>0.476</td>
</tr>
<tr>
<td>LBP + MKL(HistInt, pol=2, pol=6) [27]</td>
<td>0.670</td>
<td>0.715</td>
<td>0.571</td>
</tr>
<tr>
<td>Encoded + SVM-Linear</td>
<td>((\sigma = 0.5\sqrt{2}))</td>
<td>((\sigma = 0.5\sqrt{2}))</td>
<td>((\sigma = 0.5\sqrt{2}))</td>
</tr>
<tr>
<td>Encoded + SVM-pol=3</td>
<td>((\sigma = 1.5\sqrt{2}))</td>
<td>((\sigma = 0.5\sqrt{2}))</td>
<td>((\sigma = 0.5\sqrt{2}))</td>
</tr>
<tr>
<td>Encoded + SVM-pol=5</td>
<td>((\sigma = 1))</td>
<td>((\sigma = 0.5\sqrt{2}))</td>
<td>((\sigma = 0.5\sqrt{2}))</td>
</tr>
</tbody>
</table>

Results on the YORK and SMIC datasets are reported in Table III and IV respectively. In both datasets the encoding using the re-parametrization gives better results than the method reported in [27]. These improvement is specially observed in the results on the YORK dataset where the best recognition rates were 0.7759, 0.6667 and 0.7963 in detection, emo/~ emo and lie/truth respectively. On the SMIC dataset, the best recognition rates were 0.7759 and 0.833 in detection and neg/pos respectively.

In addition in the MKL case our method uses only one type of kernel per classifier with all the features spaces computed using the encoding at different values of $\sigma$, this means a low computational cost during the recognition process without lost of performance. In contrast, Pfister et al [27] propose a mixture of different types of kernels (Histogram intersection, Polynomial with different orders) without changing the feature space. This methodology could generate computational expensive classifiers compared with the method proposed in this paper.

V. CONCLUSIONS

We have developed a method to encode LBP using a re-parametrization of the second local Gaussian jet, arguing
that the information provided by re-parametrization can improve drastically the recognition process by correcting the statistically instability of the original LBP-TOP. Our proposed method has two main processes, the re-parametrization process which is used to compute the parameters $\theta$, $l$, $b$ and $a$ in a video sequence and the encoding process which combines the textural information provided by the LBP and the robustness of the re-parametrization. We used the proposed encoding as a solution for recognizing spontaneous micro-expressions. We tested the proposed solution on two challenging micro-expression datasets, concluding that the encoding is really promising relative to the state-of-the-art methods in micro-expressions recognition. In particular the encoding plus a Multiple Kernel Learning using linear kernels achieved a higher recognition rate in both datasets, this results is of interest from a complexity point of view because the computational cost of the linear kernel is considerable low compared with the non-linear options such as polynomial or RBF.

**Future work**

The encoding process proposed in this paper can generate high-dimensional feature sets with redundant information that could be computationally expensive in some real world tasks such as micro-expressions detection in long video sequences. In this aspect, we are currently working in a mathematical method similar to the method proposed to Lu et al. [31] to reduce the dimension of the final spatio-temporal feature set in a tensorial fashion without lost of performance. In addition, we are working in a machine learning method to select the most representative $\sigma$ value or/and the most representative encoding angle for a specific recognition task directly from the learning process.

**References**


**TABLE IV: Comparisons of Recognition rate scores of the encoding with the state-of-the-art approaches on the SMIC dataset**

<table>
<thead>
<tr>
<th>Method</th>
<th>Classes</th>
<th>detection</th>
<th>neg/pos</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP+SVM(pol=6) [27]</td>
<td></td>
<td>0.703</td>
<td>0.542</td>
</tr>
<tr>
<td>LBP+MKL(HistEnt, pol=6, pol=6) [27]</td>
<td></td>
<td>0.714</td>
<td>0.602</td>
</tr>
<tr>
<td>Encoded + LIBLINEAR ($\sigma = 0.5\sqrt{2}$) ($\sigma = 1$)</td>
<td></td>
<td>0.714</td>
<td>0.587</td>
</tr>
<tr>
<td>Encoded + SVM-Linear ($\sigma = 0.5\sqrt{2}$) ($\sigma = 0.5\sqrt{2}$)</td>
<td></td>
<td>0.702</td>
<td>0.540</td>
</tr>
<tr>
<td>Encoded + SVM-pol=3 ($\sigma = 1$) ($\sigma = 1.5\sqrt{2}$)</td>
<td></td>
<td>0.682</td>
<td>0.678</td>
</tr>
<tr>
<td>Encoded + SVM-pol=5 ($\sigma = 0.5\sqrt{2}$) ($\sigma = 0.5\sqrt{2}$)</td>
<td></td>
<td>0.723</td>
<td>0.663</td>
</tr>
<tr>
<td>Encoded + MKL(lin)</td>
<td></td>
<td>0.7426</td>
<td>0.8333</td>
</tr>
<tr>
<td>Encoded + MKL(pol=3)</td>
<td></td>
<td>0.7574</td>
<td>0.5556</td>
</tr>
<tr>
<td>Encoded + MKL(pol=5)</td>
<td></td>
<td>0.7759</td>
<td>0.6667</td>
</tr>
</tbody>
</table>