Local Patterns and Big Time Series Data for Facial Poses Classification

Hayet Mekami[1*], Sidahmed Benabderrahmane[2], Abdennacer Bounoua[1], Abdelmalik Taleb-Ahmed[3]

[1] Djillali Liabes University, Electrical Engineering Department, Bel Abbes 22000, Algeria.
[2] Paris 8 University, CS Department, LIASD, 2 Rue de la Liberté, 93526 Saint-Denis, France.
[3] Valenciennes University, Voirie Communale Université Val Mont Houy, 59300 Famars, France.

* Corresponding author. Email: hayet.mekami@univ-sba.dz
Manuscript submitted April 21, 2016; accepted January 3, 2017.
doi: 10.17706/jcp.13.1.18-34

Abstract: The problem of identifying and analyzing faces in images is a fundamental task in computer vision. Though great progress has been achieved in face detection, it is still difficult to obtain the pose estimation. In this paper we propose a pose estimation approach that is based on time series representation. We have converted input images of faces into big time series datasets, and we then used a dimensionality reduction method to convert the original series to a symbolic representation. Classification algorithms are then applied using the distances between the symbolic sequences of time series. Since external conditions when capturing images are not always optimal, pose estimation can become a challenge. In order to overcome such problems, we propose to use the gradient image and the Local Binary Pattern (LBP) combined with dynamic morphological quotient image (DMQI-LBP), where these descriptors are robust to changes in illumination. Classification algorithms such as K-means, SVM and KNN were evaluated to classify frontal vs profile faces poses, and the obtained experimental results show that the proposed method is very efficient.

Key words: Time series, machine learning, facial pose classification, image processing, data mining.

1. Introduction

Pose estimation has become one of the most attractive research topics in the field of computer vision. Indeed, for human it is easy to detect faces and know their poses, but it is difficult to do the same by a machine. It is an essential skill that is needed in face recognition, human computer interaction, faces and persons tracking. Facial pose estimation can improve the performance of face recognition system, and the choice of the frontal view reduces the search space of similarity between the facial image and the gallery images. In [1], we give an overview of different approaches for head pose estimation. Though great progress has been achieved in face detection, it is still difficult to obtain the best pose estimation. In this paper we propose a novel pose estimation approach that is based on time series representation. We have converted input images of faces to time series, and then we used the Symbolic Aggregate Approximation (SAX) technique to reduce dimensionality of data and represent the numeric time series as symbolic sequence. Classification algorithms are executed using the calculated pairwise distances between all symbolic sequences for classification purposes. Since the captured images are not always acquired in the most controlled illumination environments, we propose to use the gradient image and the Local Binary Pattern

(LBP), improved with dynamic morphological quotient image (DMQI-LBP), since these descriptors are robust against changes in illumination. The approach is efficient and easy to implement and the experimental results on several databases have shown very good classification rate.

The remainder of this paper is organized as follows. Section 2 presents the related work on face pose estimation. Section 3 and 4 describe technical details of the proposed face pose estimation method. Section 5 shows the experimental results on public databases. Section 6 concludes the paper.

2. Related Works

The head pose estimation is the most important step in diverse applications in face recognition, driver monitoring and human-computer interaction. That is why it made the subject of many recent works. The problem is how to increase the performance of any system for these applications. The pose, illumination, facial expressions, and subject variability are the factors affecting the accuracy of the process of pose estimation. Chutorian *et al*. in [1] give a good survey of the main techniques published in this domain. They classified these methods into eight categories: appearance template methods, detector array, nonlinear regression methods, manifold embedding methods, flexible models, geometric methods, tracking methods, hybrid methods. Each of these methods has its own strengths and limitations.

In the first category of methods (i.e. appearance Template methods); the images are compared with a set of templates in order to find the similarity between them [2], [3]. In this case the resolution of image is not a challenge because the template is well adapted with any changes of face. However, the process takes more time than the Detector array methods [4], [5] in which is considered each pose as a class and each class as detector, and a numerous supervised learning algorithm can be used for training the classes on many images. However, it still sensitive to non-uniform sampled training data and the variations of illuminations. Non-linear regression and Manifold methods are based all both, on the dimensionality reduction. The Non-linear regression methods look for a relationship between the mapping of features space and a space of poses in order to predict the pose of any new face.

Different regressors have been suggested. Li *et al*. [5] used Support Vector Regressors (SVRs), Gaussian Progress Regression (GPR) [6] and Convex Regularized Sparse Regression (CRSR) [7].

In Manifold embedding methods the reduction of dimension is obtained by creating a low dimensional that represents the continuous variation in head pose of the input images; than projecting the new image to the subspace and embed the features onto a pose manifold, that allows the estimation of orientation of the face [8], [9]. The results depend on the choice of the subspace, how to recover all the face pose without any other changes. For this purpose, more works are suggested e.g. PCA [10], kernel PCA (KPCA) [8], multiclass linear discriminant analysis (LDA) or the kernelized version, KLDA [11], Locally Linear Embedding (LLE) [12], and Laplacian Eigen-maps (LE) [13], Locally Embedded Analysis (LEA) [14]. Balasubramanian *et al.* [15] used an approach Based on Manifold Embededding (BME). In [16] Wang *et al*. used Isometric feature mapping (Isomap) to supply a level of supervision to traditional manifold methods through the implementation of Local Fisher Discriminant Analysis (LFDA). Ben Abdelkader [17] employed supervised variants of Neighborhood Preserving Embedding (NPE) and Locality Preserving Projection (LPP). Huang *et al*. in [18] proposed Supervised Local Subspace Learning (SL2) that builds local linear models from a sparse and non-uniformly sampled training set. Foytik *et al*. [19] used supervised class based methods for a coarse assessment of manifold locality and rely on regressive methods for creating a fine estimation of head pose. Despite all these works it is still hard to find the optimal dimensions.

Flexible models consist to use a deformable models of the human face, then to compare it with the shape and appearance of the new image to detect the location of the landmarks of this one. Among these models there are Active Shape Models (ASMs) that iteratively deform the image to fit to an example of the face in a

new image [20]. Active Appearance Models (AAMs) [21]-[23] combines the full shape model and texture variation that are learned from a training set. Constrained local models (CLMs) bring deformable model created from the local textures which are found around specific feature positions such as the eyes and mouth [24], [25]. In these methods the model adapts to the image and finds accurately the locations of feature points, which lead to a good invariance to head localization. As well as provide a high efficiency, especially for the head orientation where the outer corners of eyes are visible.

Geometric methods are based on the identification of the positions of facial features such as eyes, mouth, nose, from their location on a chosen geometry to estimate the pose. Gee *et al*. [26] use nose tip and far corners of the eyes and mouth to estimate face pose. But generally in facial images, it is not easy to localize the nose precisely. In [27], [28] a triangle made by the centers of the two eyes and mouth has a distinct shape for estimate the face pose, by consequence the error in the location of these features degrade the performance. In tracking methods, for estimation the pose the different changes of image in consecutive frames of a video sequence are considered as criteria [29]. Hybrid methods combine one or more of different approaches to compensate the disadvantages [30].

Although there are many approaches and methods suggested to determine the pose of face in the image, the problem still exists where there are several disadvantages and limitations to each method used. Such as complexity of the models, sensitivity to noise and to images acquired in degraded conditions, and low speed of processing.

The methods based on features, like flexible models, geometric and tracking methods, have a good invariance to head localization. But, these methods have the disadvantage, so that the facial feature locations should specify in all images manually in advance to make the training dataset. They are also computationally expensive.

The approaches based on appearance such as template methods, detector array, nonlinear regression and manifold embedding methods, are altogether efficient in regards to computation time. Therefore, they have attracted more and more attentions especially the non-linear regression and manifold embedding techniques which have been extensively used at the recent study for head pose estimation [31]-[34]. The complexity of the non-linear and linear mappings of the facial images and pose labels make difficult to develop an exact function for robust head pose estimation. Also, it still helpless to effectively model the structure of the subspace.

To address these limitations, we propose a new approach based on the use of dimensionality reduction with time series. The advantage of the approach is the easy way of the representation of the initial numeric matrix of the image as symbolic values, thus allowing us to avoid problems of noise, variation of illumination...etc.

The other major point is the ability of using powerful symbolic data mining techniques to classify faces poses of any dataset, and thus efficient symbolic distances for classification purposes.

3. Time Series Representations

Times series (TS) techniques have been widely used during the two last decades. Time series data are occurring almost everywhere in various domains from medical [35] (electrocardiogram ECG, blood pressure), satellite image [36], finance and business [37] (stock market, profit-and-loss of a company etc.), meteorology (variation in temperature or pressure or wind speed daily, monthly, or yearly), entertainment (music, movies), sociology (crime figures number of arrests, etc.) [38], bioinformatics, pattern recognition, text mining [39], computer vision, … etc.

In practice, numerical TS suffer from the high dimensionality, which is not convenient in the storage of this kind of data and the computational complexity when manipulating theme. These difficulties led us to

propose solutions involving dimensionality reduction of the data. Several time series based data mining algorithms use representations of reduced dimensionality. Among them: Discrete Fourier transform (DFT) [40], Discrete Wavelet Transform (DWT) [41], piecewise linear (Piecewise Aggregate Approximation PAA) [42], piecewise constant adaptive (Adaptive piecewise constant approximation APCA) [42], [43], and singular value decomposition (SVD singular Value Decomposition) [43]. Indeed, the use of these representations reduces the dimension but its most inconvenience is the fact that the distance between sequences has low correlations to the distance defined between the original time series (numerical values usually the Euclidean distance). To overcome this problem Lin *et al*. proposed in [44] an approach called Symbolic Aggregate Approximation (SAX). SAX representation solves both dimensionality reduction and the lower bounding for Euclidean distance with a very simple proposed distance [44].

3.1. Symbolic Aggregate Approximation (SAX)

The SAX method is a symbolic representation of time series with a dimensionality reduction and a lower bound of the Euclidean distance. SAX algorithm has three main steps to transform the TS form n dimensions to w dimension ($w \ll n$). The time series itself must be at first normalized to achieve a mean of zero and a standard deviation of one. Then, the original *TS* is transformed into PAA (Piecewise Aggregate Approximation), while the data is divided into *w*} segments with equal length (frame also known as codeword (*w*)) and the average value of each codeword is calculated and a vector of these values becomes the data reduced representation. Next, from the "breakpoints" that divide the distribution space into a equiprobable regions are determined. Breakpoints are a sorted list of numbers $B = \beta 1 \ldots \beta a\text{-}1$ such that the area under a $N(0; 1)$ Gaussian curve from $\beta_i - \beta_{i-1} = 1/a$ (a is alphabet size also known as codebook) [44]. A lookup table that contains the breakpoints is shown in Table 1.

Table 1. Lookup Table That Contains Statistical Breakpoints [44]

a	3	4	5	6
β_1	-0.43	-0.63	-0.84	-0.97
β_2	0.43	0	-0.25	-0.43
β_3		0.63	0.25	0
β_4			0.84	0.43
β_5				0.97

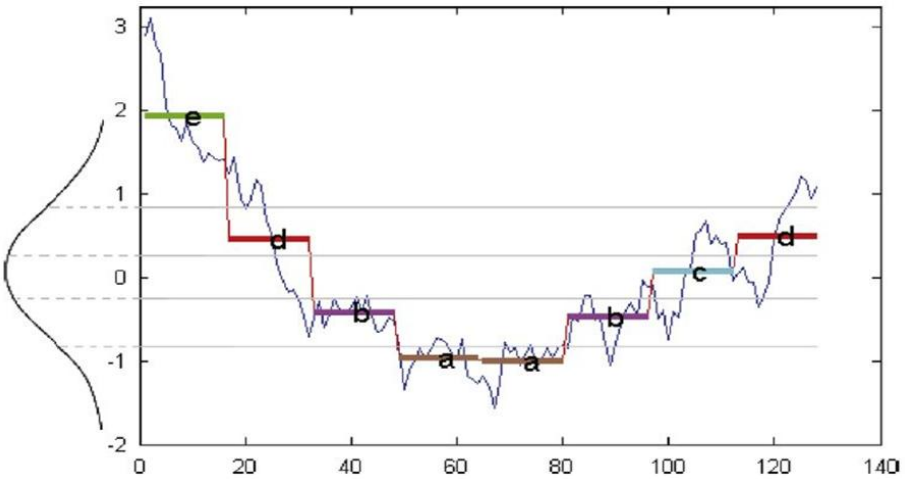


Fig. 1. Example of SAX representation of a time series with the number of segments w equal to 8 and the size of alphabetic symbols a is 5.

Finally, the PAA representation is symbolized into a sequence of discrete string. The interval between two successive breakpoints is assigned to a symbol of the alphabet, and each segment of the PAA within this interval is discretized by this symbol. So all PAA coefficients that are below the lowest breakpoint are

encoded by the symbol "*a*", then all PAA coefficients that are above or equal the lowest breakpoint and lower than the second smallest breakpoint are encoded by the symbol " *b*", the following symbol is "*c*" etc.

Then all PAA coefficients that are above or equal the lowest breakpoint and lower than the second smallest breakpoint are encoded by the symbol "*b*", the following symbol is "*c*" etc.

Measuring the similarity is important task of the time series data mining. One of the most positive aspects of SAX, is that it represents lower bounding for Euclidean distance. To measure the similarity, they use the following formula:

$$MINDIST(\hat{Q},\hat{C}) \equiv \sqrt{\frac{n}{w}}\sqrt{\sum_{i=1}^{w} dist(\hat{q}_i - \hat{c}_i)^2} \quad (1)$$

where $\hat{Q}$ and $\hat{C}$ are the symbolic representation of numerical time series Q and C respectively. The "*dist*" function is implemented using the lookup table for the particular set of the breakpoints as illustrated in Table 2 [44]. The distance "dist(*r,c*)", between two SAX symbol values r and c is calculated by the following expression:

$$dist(r,c) = \begin{cases} 0 & \text{if} |r-c| \le 1 \\ \beta_{\max(r,c)-1} - \beta_{min(r,c)} & \text{otherwise} \end{cases} \quad (2)$$

Thus, the distance between any successive symbols of the alphabet is zero.

Table 2. A Lookup Table Used by the MINDIST Function for an Alphabet Size a = 4 [44]

	a	*b*	*c*	*d*
a	0	0	1,34	0,67
b	0	0	0	0,67
c	0,67	0	0	0
d	1,34	0,67	0	0

4. Overview of Our Approach

As reported above, the main objective of our work is to estimate the pose of the face in an input image, using time series representation so that the dimensionality is reduced and consequently the complexity of the learning and time become low.

Initially, the matrix representing the input image is converted to a vector which is considered as time series. In order to make faster the step of time series transformation, we have scanned the image line by line from left to right starting from the top left as Fig. 2 shows.

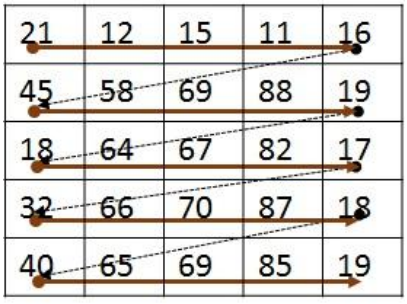


Fig. 2. The conversion of an image into 1D time series sequence.

After that, a transformation is applied to each generated time series by applying SAX symbolization technique which encodes the numerical series to a symbolic sequence.

After this step, and having a set of symbolic time series that represent all the images in a learning dataset, a similarity matrix is produced by calculating all the pairwise "*MINDIST* "distances between all the sequences. Finally, classification algorithms are used to decide whether a face is in frontal view or not. Fig. 3 presents the general steps of the approach.

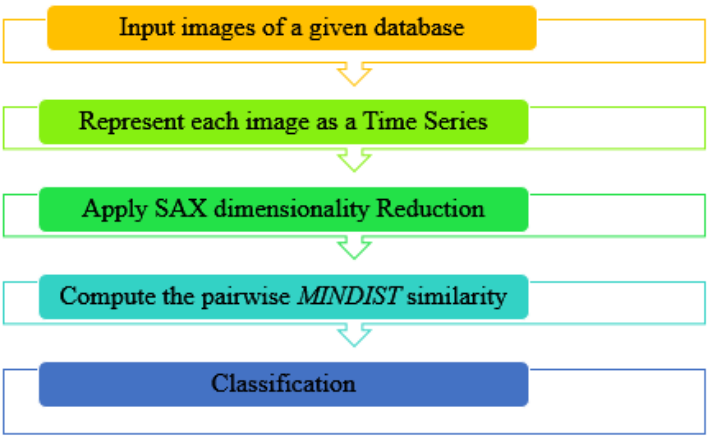


Fig. 3. The general steps of the approach.

Note that the input images are cropped manually of each facial image with different poses, and are resized to size 100*100 then converted to gray-scale. Thus the length of the numerical time series is 10000 (the total number of the pixels in image 100×100). Since the learning datasets are huge we adapted the classification algorithms on hadoop map reduce.

4.1. Symbolic Aggregate Approximation (SAX)

In order to transform the numerical time series of the image into a symbolic sequence, we used SAX approach [40]. SAX reduces the dimension of multivalued TS using the mean of the data on a sliding windows. Fig. 4 illustrates the principle of the approach. As detailed above, at each position of a frame (sliding window) the mean value is calculated over the points in that position, and a symbol is associated to the mean values at each window position by a codebook learning process. The symbols are defined by the user as the Gaussian quantiles points. SAX requires a good choice of parameters for efficient symbolic representation of the original TS to their SAX representation. Therefore, to determine the best size of the frame and the alphabet, we performed several experiments, by changing the frame size from 5 to 64 and size of alphabet 5 to 128. In each SAX resolution (frame size x alphabet size), we used k-means with map-reduce version since the size of data is huge, SVM, KNN algorithms to classify the faces images in frontal vs non frontal poses. The results of our approach evaluation are shown in the next section.

4.2. Images Filtering

As in the real cases, the images are with different illumination; hence we use three categories of images during the learning of the model with three categories of images. Firstly, we use the images of the database as they are without processing. Secondly, we apply a gradient filter on images. Then, we use the LBP transformation of the image quotient filtered by a morphological filter.

The gradient of an image measures the changes in the intensity of the same point in the original image in the horizontal and vertical directions. Mathematically, the gradient of a function of two variables is a vector in two dimensions. The modulus of the vector is the magnitude of the gradient which tells us how quickly the image is changing, while the direction of the vector tells us the direction in which the image is changing most rapidly. Fig. 5 shows that the gradient of the original image in low lighting is almost the same as the image in natural lighting, because the gradient operator is not very sensible to illumination changes,

therefore this allows us to increase the rate of classification. Because the gradient acts as a high-pass filter which renders it sensitive to noise, so the image gradient is filtered by Gaussian filter with Unsharp contrast enhancement filter which sharpened edges of the elements without increasing noise or blemish.

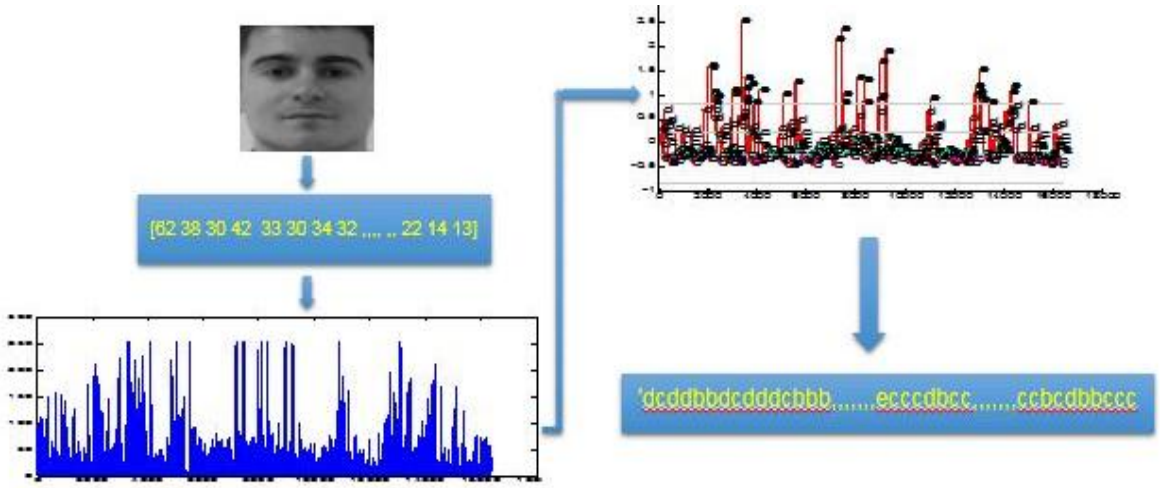


Fig. 4. Conversion of an image in a SAX sequence. Having a learning dataset of images, they will be all converted to a collection of SAX sequences.

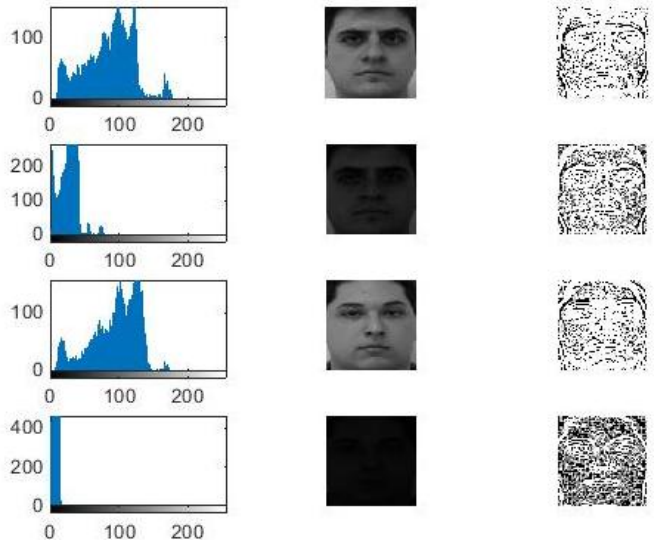


Fig. 5. Example the gradient of the original image in low and natural illumination.

Then we used the DMIQ-LBP image, as shown in Fig. 6 which is obtained after the application of Dynamic Morphological Quotient Image [45], combined with LBP (Local binary pattern) [46].

An image in certain lighting conditions can be represented by the produce of the illumination L and reflectance R. Such a module can be expressed as follows:

$$I(x, y) = L(x, y) * R(x, y) \tag{3}$$

where $I(x, y)$ is a value of each pixel in image, $L(x,y)$ is dependent of the lighting source, while $R(x,y)$ is determined by the characteristics of the surface of object.

Using equation (3), the reflectance can be expressed as the quotient of the base image and the illumination L [47]. The filtering of the illumination will lead to invariance of the reflectance. A convenient filter can reach this aim. Motivated by the low complexity and the good performance of the morphological quotient image (MQI), the estimation of the illumination $L(x,y)$ is done by using a morphological close operator, which is a non-linear operator defined by a dilation followed by an erosion.

$$R(x,y) = \frac{I(x,y)}{L(x,y)} = \frac{I(x,y)}{Close(x,y)} \tag{4}$$

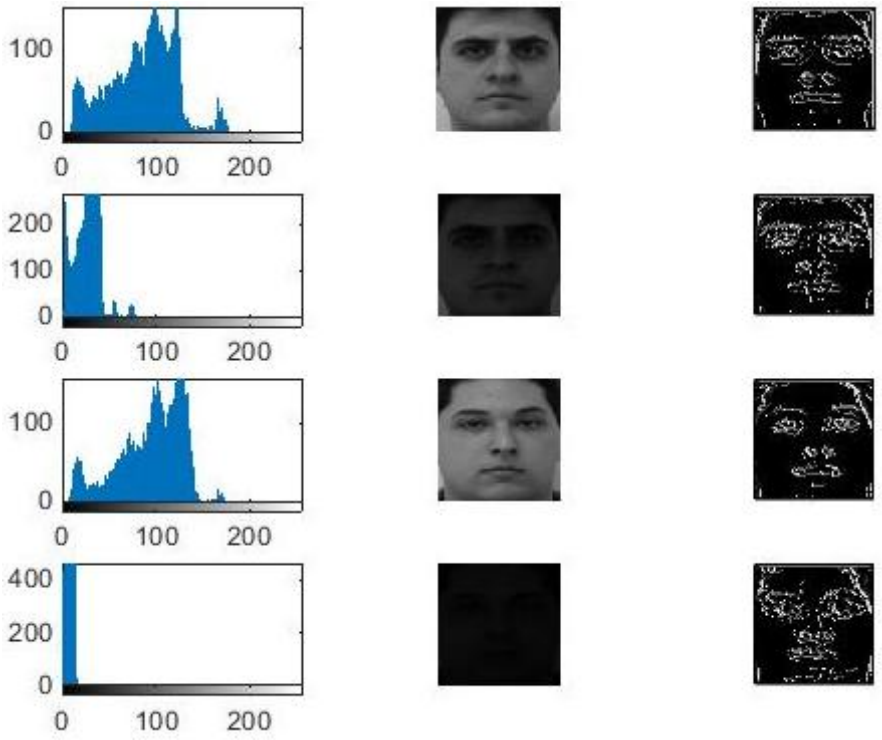


Fig. 6. Example DMQI-LBP image of the original image in low and natural illumination.

The dilatation effect is to expand the image where the pixels of the expanded image are the sum of the pixels of the original image and the structuring element. This transformation tends to eliminate dark objects. Contrariwise, erosion is the effect of shrinking the image, where the pixels of the eroded image are the difference pixels of the original image and the structuring element. Erosion allows darken and spread the edges of the darkest objects. Therefore, with a suitable size of a structuring element the close operation can preserve some particulate pattern while it attenuated other. The close eliminates the dark areas that are smaller than the structuring element, keeps the edges of the object and connect the areas of the same light intensity. The way to make the illumination invariant is to use the close operator, which lead to a smooth version of the input image especially for images with low lighting.

The size of the structuring element pays an important role for a good morphological filter. Wang *et al.* [45] have indicated that with a large structuring element the close operator keeps only the large scale features, but poor performance to compensate on local illumination especially in the case of images under dark zone. On the other hand, with a small size, it results in good local illumination normalization, but simultaneously misses large scale features. To overcome this problem Zhang *et al.* [48] proposed Dynamic Morphological Quotient Image (DMQI) using a structuring element with dynamically size according to the formula 5. DMQI is expressed by the equation 6.

$$DClose(x,y) = \begin{cases} Close^{l}(x,y) & Close^{l}(x,y) > \alpha.Close^{s}(x,y) \\ Close^{m}(x,y) & \alpha.Close^{s}(x,y) > Close^{l}(x,y) > \beta.Close^{s}(x,y) \\ Close^{s}(x,y) & \beta.Close^{s}(x,y) > Close^{l}(x,y) \end{cases} \tag{5}$$

$$DMQI(x,y) = \frac{I(x,y)}{L(x,y)} = \frac{I(x,y)}{DClose(x,y)} \tag{6}$$

where α and β are the parameters of the feature scales, while $\alpha>\beta>1.0$. l, m, and s are the optional sizes of templates, while $l > m > s > 1$ [48].

In the regions of brow, eye, nose, mouth, or the boundary of changing light intensity, the grayscale is changing significantly. In this case the choice of close operator with large size is better to keep the features of these regions. So the DMQI image is calculated using the condition $Close^l(x,y) > \alpha . Close^s(x,y)$, which shows that pixels of close operator with a large size is very different than the pixels of image obtained by close operator with small size.

However, if the regions are under illumination or in a smooth region, such as cheek and forehead, the change of gray values in these regions is weak. So, in this case the use of close operator with small size is sufficient. Thus, DMQI image is calculated using the condition $\beta . Close^s(x,y) > Close^l(x,y)$ [48].

5. Experiments and Results

To evaluate the performance of our approach, we tested it on three databases, namely GTAV [49], FEI [50] and FERET [51]. The first database (GTAV) includes a set of 44 persons with 27 pictures per person which correspond to different pose views (0°, ±30°, ±45°, ±60° and ±90°) under three different illuminations (environment or natural light, strong light source from an angle of 45°, and finally an almost frontal mid-strong light source. Fig. 7 presents some example of this database.

Fig. 7. Example of a face from the GTAV Face database (different illuminations).

The second database (FEI) contains 2800 images of 100 men and 100 women; each individual has 14 images in an upright frontal position with profile rotation of up to about 180°. In Fig. 8 some example of this dataset. It has 400 frontal images in natural light and 400 images in low lighting. In FEI database each person has 12 images in different pose with natural light and two images in frontal view with weak light.

Fig. 8. Some examples of image variations from the FEI face database.

Fig 9. Example of a person from the FERET Face database.

The third FERET database consists of images that are collected in a semi-controlled environment, of different age, race, and sex distribution. With poses fa, fb for frontal pose, and ql, qr for the left and right respectively quarter pose (±22,5°), hl, hr are the poses mid-left and mid-right respectively (±67.5°), and pl,

pr are profile poses left and right respectively (±90 $°). Fig. 9 shows some example of this dataset. The global total of the used image is around 9180 images.

In this section, we present the results of the experimental validation. They are reported for each database independently.

We recall that at the learning step, they encode each image in a SAX symbolic time series. Since SAX requires two parameters, namely the size of the windows frame and the length of the alphabet, we performed as a first experiment, a tuning procedure using GTAV database to find out the best parameters "*w*" and "*a*" of SAX, that maximize the classification rate.

We have conducted the experiments using the images as they are without any modification or processing, and we have applied our algorithm on this database.

We calculated the classification rates with different values of "*w*" and "*a*".

In Table 3, we show the F-Score of classification for each frame size (*w*=5, 6, 7, 8, 9, 10, 15, 20, 25, 35, 45, 55 and 64) and with different alphabet size (*a*=5, 6, 7, 8, 64, 128). We classified the poses using K-means algorithm into three main classes: class of frontal pose, class of the left view, and class of the right view that group poses in quarter profile (left or right) at full profile. We can observe that with SAX, and when maximizing the size of the window, the classification rate decreases. This is totally normal since the SAX symbolic encoding is lossless with great values of codeword (*w*).

In order to ensure the best classification rates, we choose the smallest frame size (*w*=5), and applied it at the rest of the evaluations to represent the time series in the next experiments.

After determining the frame size, we should determine the best alphabet size. We have fixed *w* at 5, while the alphabet size "*a*" varies in 5, 10, 15, 20, 64, and 128. To evaluate the classification rate, we have performed experiments using K nearest neighbor(K-NN) and support vector machine (SVM) with Gaussian kernel function for each database.

These classification algorithms were repeated with the three categories of images: without filter (noted OUTPRS in the tables), with Gradient filter (noted GRAD in the tables), and with DMQ-LBP (noted DMQLBP in the tables).

The classification results of GTAV database are listed in Tables 4 and 5. We can resume from these tables that:

- Frontal poses have reached a classification rate of 100% (*w*=5, *a*=10...64 with gradient, *a*=5, 128 with DMQ-LBP using KNN; and *a*=5, 10 without processing, *a*=10, 15, 64, 128 with Gradient, *a*=5, 10, 15, 128 with DMQ-LPB using SVM).
- For left and right classes almost poses have been well classified by the three approaches.
- The classification with SVM algorithm allows us to achieve the best classification rate comparing to KNN algorithm.
- Using images without processing all poses in frontal view were classified correctly (with SVM).
- Using Gradient and DMQ-LBP, all poses were nearly classified successfully for each alphabet size.

Table 3. F-Score of the Classification of Faces Poses on GTAV Database Using K-means Algorithm to Evaluate the Parameter *w* (Frame Size)

Size Frame	Size Alphabet	View Left %	Frontal %	View Right %
5	5	81,76	45,70	87,24
	6	82,92	48,02	87,95
	7	82,02	47,22	88,55
	8	81,15	46,76	88,90
	64	86,32	54,11	90,60
	128	86,92	54,70	90,60
10	5	81,80	45,22	86,69
	6	82,19	46,18	87,40
	7	82,17	45,54	87,51

	8	81,33	46,30	88,21
	64	82,25	48,53	89,14
	128	82,25	48,72	89,37
15	5	81,70	44,76	88,16
	6	81,84	46,22	88,34
	7	81,67	46,52	88,57
	8	80,75	46,15	88,91
	64	80,31	46,99	89,18
	128	80,97	47,45	89,18
20	5	79,70	39,14	85,86
	6	80,90	40,08	86,19
	7	81,33	41,20	86,28
	8	80,39	42,44	87,24
	64	80,62	45,53	88,80
	128	81,72	46,55	88,87
25	5	79,96	40,94	86,78
	6	81,63	42,51	87,10
	7	81,74	44,00	87,68
	8	80,84	44,01	87,62
	64	81,55	46,85	89,09
	128	81,55	46,55	89,00
35	5	78,63	40,39	87,70
	6	78,62	42,39	87,82
	7	78,62	42,39	87,82
	8	76,96	42,13	88,44
	64	77,92	43,43	88,64
	128	77,97	43,59	88,64
45	5	79,57	40,74	88,26
	6	77,49	41,35	88,18
	7	78,44	41,49	87,89
	8	78,68	39,45	87,00
	64	77,70	41,92	87,83
	128	77,29	41,68	87,83
55	5	78,38	40,00	87,65
	6	77,17	41,50	88,29
	7	78,44	41,49	87,89
	8	76,40	39,15	87,45
	64	76,35	39,23	87,02
	128	76,44	39,69	86,84
64	5	78,23	40,98	88,31
	6	76,27	39,84	87,98
	7	78,19	39,28	86,93
	8	76,42	39,36	87,75
	64	76,01	38,30	86,29
	128	76,01	38,30	86,29

Table 4. Classification Rate of the GTAV Database for Each Approach Using K-NN Algorithm to Determine the Best Alphabet Size

Size Frame	Size Alphabet	View Left %	Frontal %	View Right %	
5	5	99,72	97,4	99,43	O
	10	99,34	96,32	98,96	U
	15	99,24	95,88	99,34	T
	20	99,62	98,15	99,90	P
	64	99,62	98,89	99,53	R
	128	99,43	96,63	99,34	S
5	5	99,81	99,62	99,90	
	10	100	100	100	G
	15	100	100	100	R
	20	99,91	100	99,91	A
	64	100	100	100	D
	128	99,91	99,63	100	
5	5	100	100	100	D
	10	99,53	98,50	99,15	M
	15	99,53	99,26	99,71	Q

	20	99,71	99,63	99,81	L
	64	99,62	99,62	99,72	B
	128	99,81	100	99,81	P

Table 5. Classification Rate of the GTAV Database for Each Approach Using SVM Algorithm to Determine the Best Alphabet Size

Size Frame	Size Alphabet	View Left %	Frontal %	View Right %	
5	5	99,72	100	99,72	O
	10	99,72	100	99,72	U
	15	99,81	99,63	99,90	T
	20	99,25	99,62	99,34	P
	64	99,24	99,26	99,43	R
	128	99,81	99,81	99,63	S
5	5	99,90	100	99,90	
	10	100	100	100	G
	15	99,90	100	99,90	R
	20	99,90	98,89	99,81	A
	64	100	100	100	D
	128	99,90	100	99,90	
5	5	99,90	100	99,90	D
	10	99,81	100	99,81	M
	15	99,81	100	99,81	Q
	20	99,72	99,25	99,90	L
	64	99,90	99,26	99,72	B
	128	100	100	100	P

In Tables 6 and 7 we illustrate similar results using K-NN and SVM on FET database. The frontal poses were classified with the rate between 97% and 98%. Almost all the poses from left or right are well classified.

Table 6. Classification Rate of the FEI Database for Each Approach Using K-NN Algorithm to Determine the Best Alphabet Size

Size Frame	Size Alphabet	View Left %	Frontal %	View Right %	
5	5	98,44	97,99	98,56	O
	10	98,32	97,91	98,55	U
	15	98,19	97,60	98,26	T
	20	98,31	97,96	98,62	P
	64	98,26	97,67	98,22	R
	128	97,70	97,36	98,12	S
5	5	98,32	97,74	98,31	
	10	98,31	97,92	98,56	G
	15	98,94	98,24	98,45	R
	20	98,01	97,61	98,44	A
	64	98,56	97,91	98,32	D
	128	98,31	97,99	98,31	
5	5	98,25	98,04	98,81	D
	10	98,11	97,68	98,23	M
	15	98,00	98,08	98,93	Q
	20	98,81	98,81	98,38	L
	64	98,05	97,92	98,81	B
	128	98,69	98,28	98,75	P

Table 7. Classification Rate of the FEI Database for Each Approach Using SVM Algorithm to Determine the Best Alphabet Size

Size Frame	Size Alphabet	View Left %	Frontal %	View Right %	
5	5	98,38	97,83	98,37	O
	10	97,46	97,19	98,3	U
	15	98,70	97,80	98,00	T
	20	97,71	97,01	97,88	P
	64	98,38	97,73	98,38	R
	128	97,89	97,31	98,13	S
	5	97,94	97,83	98,68	G

5	10	97,53	97,35	98,43	R
	15	98,38	98,21	98,93	A
	20	97,33	96,85	98,01	D
	64	97,20	97,11	98,38	
	128	98,39	98,45	99,06	
5	5	98,75	97,83	97,75	D
	10	98,82	98,91	99,31	M
	15	98,76	98,66	98,75	Q
	20	98,75	98,79	98,81	L
	64	98,57	97,66	97,93	B
	128	98,88	98,33	99,12	P

Table 8. Classification Rate of the FERT Database for Each Approach Using K-NN Algorithm to Determine the Best Alphabet Size

Size Frame	Size Alphabet	View Left %	Frontal %	View Right %	
5	5	99,39	98,87	99,41	O
	10	98,33	97,15	98,38	U
	15	98,41	97,29	98,28	T
	20	98,71	97,63	98,50	P
	64	98,71	97,46	98,52	R
	128	98,72	97,29	98,52	S
5	5	86,83	90,59	90,40	
	10	86,86	91,00	90,20	G
	15	86,86	91,00	90,20	R
	20	86,12	90,78	89,60	A
	64	87,73	91,74	90,38	D
	128	86,43	91,75	89,91	
5	5	99,39	98,89	99,19	D
	10	99,24	98,87	99,26	M
	15	99,16	98,70	99,23	Q
	20	99,36	98,83	99,38	L
	64	99,30	98,76	99,17	B
	128	99,19	98,74	99,35	P

Table 9. Classification Rate of the FERT Database for Each Approach Using SVM Algorithm to Determine the Best Alphabet Size

Size Frame	Size Alphabet	View Left %	Frontal %	View Right %	
5	5	99,42	99,15	99,46	O
	10	99,13	98,61	99,18	U
	15	99,36	98,85	99,22	T
	20	99,21	98,48	99,00	P
	64	99,22	98,85	99,38	R
	128	99,16	98,24	98,76	S
5	5	84,21	89,12	92,74	
	10	75,61	87,38	86,04	G
	15	81,74	87,37	91,13	R
	20	81,71	87,40	91,16	A
	64	77,44	87,37	87,46	D
	128	82,38	87,69	91,69	
5	5	99,38	99,42	99,48	D
	10	99,35	99,28	99,34	M
	15	99,33	98,89	99,28	Q
	20	99,27	98,98	99,11	L
	64	99,36	99,41	99,46	B
	128	99,16	98,87	99,25	P

It can be deduced from these results, that if the images are under a natural environment, it is sufficient to apply SAX on time series of images with any treatment (FERT case), even in the case of images with different lighting (GTAV case). In case where the images are in weak or dark light (FEI case) it is preferable to use the images processed following the protocol of the second treatment (Gradient), or the third processing (DMQ-LBP) with high alphabet size. Therefore, the conditions in which the images were taken

are used to determine the necessary parameters to use the SAX encoding process. It is also clear that SVM classifiers are very efficient regarding k-means or KNN in our study.

6. Conclusion

In this paper, we presented a new technique for facial pose classification characterized by its simplicity, its speed in computation, and its robustness. The method uses dimensionality reduction through time series representation of the learning images. Each time series is encoded with SAX symbolic representation to transform the numerical series to a symbolic sequence with different frames and alphabet sizes.

After that, we calculated the pairwise similarity matrices between images of different databases using an adapted distance.

Several classifications methods throughout the generated big data sets of similarity matrices were used and very efficient results were obtained. The results have shown that our approach is robust and allows us to separately classify the poses even in degraded conditions. In our approach, we have reduced the space from 2D image to 1D representation by time series representation, thus with this approach we can assert that the time processing is considerably optimized.

As perspective, we would like to explore other symbolic transformation techniques such as vector quantization [52], and subsequently other numerical and / or semantic distances measures [53]. In this work we used the k-means classification algorithm under hadoop map-reduce [54]. We would like to explore the recently proposed classification algorithms which have been proposed under Spark for comparison purposes.

Acknowledgment

We want to thank Mr. Ivashko Evgeny senior research associate in the Laboratory for Telecommunications Systems Institute of Applied Mathematical Research (Russian Academy of Sciences, Moscow), for their advice and counselling.

Availability

The implemented sources of our time series encoding algorithms as well as classification are available at: https://www.dropbox.com/home/SAX. For any use or distribution purposes, please contact the authors.

References

[1] Murphy-Chutorian, E., & Trivedi, M. M. (2009). Head pose estimation in computer vision: A survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *31(4)*, 607-626.

[2] Niyogi, S., & Freeman, W. T. (1996, October). Example-based head tracking. *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition* (pp. 374-378). IEEE.

[3] Beymer, D. J. (1994). Face recognition under varying pose. *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 756-761).

[4] Viola, M., Jones, M. J., & Viola, P. (2003). Fast multi-view face detection. *Proceedings of Computer Vision and Pattern Recognition*.

[5] Li, Y., Gong, S., Sherrah, J., & Liddell, H. (2004). Support vector machine based multi-view face detection and recognition. *Image and Vision Computing*, *22(5)*, 413-427.

[6] Ranganathan, A. A., & Yang, M. H. (2008). Online sparse matrix gaussian process regression and vision applications. *Proceedings of Computer Vision–ECCV 2008* (pp. 468-482). Springer Berlin Heidelberg.

[7] Ji, H., Liu, R., Su, F., Su, Z., & Tian, Y. (2011, September). Robust head pose estimation via convex regularized sparse regression. *Proceedings of 2011 18th IEEE International Conference on Image*

Processing (pp. 3617-3620). IEEE.

[8] Chen, L., Zhang, L., Hu, Y., Li, M., & Zhang, H. (2003, October). Head pose estimation using fisher manifold learning. *Proceedings of AMFG* (pp. 203-207).

[9] Raytchev, B., Yoda, I., & Sakaue, K. (2004, August). Head pose estimation by nonlinear manifold learning. *Proceedings of the 17th International Conference on Pattern Recognition* (Vol. 4, pp. 462-466). IEEE.

[10] Srinivasan, S., & Boyer, K. L. (2002, August). Head pose estimation using view based eigenspaces. 40302. IEEE.

[11] Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern classification*. John Wiley & Sons.

[12] Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290(5500)*, 2323-2326.

[13] Belkin, M., & Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, *15*(6), 1373-1396.

[14] Fu, Y., & Huang, T. S. (2006, April). Graph embedded analysis for head pose estimation. *Proceedings of 7th International Conference on Automatic Face and Gesture Recognition* (p. 6). IEEE.

[15] Balasubramanian, V. V. N., Ye, J., & Panchanathan, S. (2007, June). Biased manifold embedding: A framework for person-independent head pose estimation. *Proceedings of CVPR'07. IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-7). IEEE.

[16] Wang, X., Huang, X., Gao, J., & Yang, R. (2008). Illumination and person-insensitive head pose estimation using distance metric learning. *Proceedings of Computer Vision–ECCV 2008* (pp. 624-637). Springer Berlin Heidelberg.

[17] BenAbdelkader, C. (2010). Robust head pose estimation using supervised manifold learning. *Proceedings of Computer Vision–ECCV 2010* (pp. 518-531). Springer Berlin Heidelberg.

[18] Huang, D., Storer, M., De la Torre, F., & Bischof, H. (2011, June). Supervised local subspace learning for continuous head pose estimation. *Proceedings of 2011 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2921-2928). IEEE.

[19] Foytik, J., & Asari, V. K. (2013). A two-layer framework for piecewise linear manifold-based head pose estimation. *International Journal of Computer Vision*, *101(2)*, 270-287.

[20] Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1995). Active shape models-their training and application. *Computer Vision and Image Understanding*, *61(1)*, 38-59.

[21] Cootes, T. F., Edwards, G. J., & Taylor, C. J. (2001). Active appearance models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, *(6)*, 681-685.

[22] Matthews, I., & Baker, S. (2004). Active appearance models revisited. *International Journal of Computer Vision*, *60(2)*, 135-164.

[23] Gao, X., Su, Y., Li, X., & Tao, D. (2010). A review of active appearance models. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *40(2)*, 145-158.

[24] Cristinacce, D., & Cootes, T. (2008). Automatic feature localisation with constrained local models. *Pattern Recognition*, *41(10)*, 3054-3067.

[25] Wang, Y., Lucey, S., & Cohn, J. F. (2008, June). Enforcing convexity for improved alignment with constrained local models. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-8). IEEE.

[26] Gee, A., & Cipolla, R. (1994). Determining the gaze of faces in images. *Image and Vision Computing*, *12(10)*, 639-647.

[27] Mekami, H., & Benabderrahmane, S. (2010, October). Towards a new approach for real time face detection and normalization. *Proceedings of 2010 International Conference on Machine and Web Intelligence* (pp. 455-459). IEEE.

[28] Shafi, M., & Chung, P. W. H. (2010). Face pose estimation from eyes and mouth. *International Journal of Advanced Mechatronic Systems*, *2(1-2)*, 132-138.

[29] Mikami, D., Otsuka, K., & Yamato, J. (2009, June). Memory-based particle filter for face pose tracking robust under complex dynamics. *Proceedings of Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (pp. 999-1006). IEEE.

[30] Vatahska, T., Bennewitz, M., & Behnke, S. (2007, November). Feature-based head pose estimation from images. *Proceedings of 7th IEEE-RAS International Conference on Humanoid Robots* (pp. 330-335). IEEE.

[31] Wang, Q., Wu, Y., Shen, Y., Liu, Y., Lei, Y. (2015). Supervised sparse manifold regression for head pose estimation in 3D space. *Signal Processing, 112*, 34-42.

[32] Liu, Y., Wang, Q., Jiang, Y., & Lei, Y. (2014). Supervised locality discriminant manifold learning for head pose estimation. *Knowledge-Based Systems, 66*, 126-135.

[33] Wang, C., & Song, X. (2014). Robust head pose estimation via supervised manifold learning. *Neural Networks, 53*, 15-25.

[34] Hu, C., Gong, L., Wang, T., Liu, F., & Feng, Q. (2014). An effective head pose estimation approach using Lie Algebrized Gaussians based face representation. *Multimedia Tools and Applications, 73(3)*, 1863-1884.

[35] J. Wang, J., Liu, P., She, M. F., Nahavandi, S., & Kouzani, A. (2013). Bag-of-words representation for biomedical time series classification. *Biomedical Signal Processing and Control*, *8(6)*, 634-644.

[36] Petitjean, F., Inglada, J., & Gançarski, P. (2012). Satellite image time series analysis under time warping. *Geoscience and Remote Sensing, IEEE Transactions on*, *50(8)*, 3081-3095.

[37] Tsay, R. S. (2005). *Analysis of Financial Time Series*, *543.* John Wiley & Sons.

[38] Greenberg, D. F. (2001). Time series analysis of crime rates. *Journal of Quantitative Criminology*, *17(4)*, 291-327.

[39] Zhao, Y. (2013). Analysing twitter data with text mining and social network analysis. *Proceedings of the 11th Australasian Data Mining and Analytics Conference (AusDM 2013)*.

[40] Faloutsos, C., Ranganathan, M., & Manolopoulos, Y. (1994). *Fast Subsequence Matching in Time-Series Databases*, *23(2)*, 419-429. ACM.

[41] Chan, K. P., & Fu, A. W. C. (1999, March). Efficient time series matching by wavelets. *Proceedings of 15th International Conference on Data Engineering,* (pp. 126-133). IEEE.

[42] Keogh, E., Chakrabarti, K., Pazzani, M., & Mehrotra, S. (2001). Locally adaptive dimensionality reduction for indexing large time series databases. *ACM SIGMOD Record*, *30(2)*, 151-162.

[43] Geurts, P. (2001). Pattern extraction for time series classification. *Principles of Data Mining and Knowledge Discovery* (pp. 115-127). Springer Berlin Heidelberg.

[44] Lin, J., Keogh, E., Lonardi, S., & Chiu, B. (2003, June). A symbolic representation of time series, with implications for streaming algorithms. *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery* (pp. 2-11). ACM.

[45] Wang, J., Wu, L., He, X., & Tian, J. (2007, September). A new method of illumination invariant face recognition. In *icicic* (p. 139). IEEE.

[46] Huang, D., Shan, C., Ardabilian, M., Wang, Y., & Chen, L. (2011). Local binary patterns and its application to facial image analysis: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 41*(6), 765-781.

[47] Shashua, A., & Riklin-Raviv, T. (2001). The quotient image: Class-based re-rendering and recognition with varying illuminations. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(2)*, 129-139.

[48] Tian, J. Y., He, X., & Yang, X. (2007). MQI based face recognition under uneven illumination. *Advances in Biometrics* (pp. 290-298). Springer Berlin Heidelberg.

[49] Tarrés, F., & Rama, A. GTAV Face Database. From http: //gps-tsc.upc.es/GTAV/ResearchAreas/UPCFaceDatabase/GTAVFaceDatabase.htm

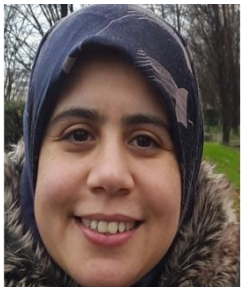
[50] Thomaz, C. E. (2012). FEI face database. Retrieved October 2, 2012, from http://fei. edu. br/~cet/facedatabase. html

[51] Phillips, P. J., Wechsler, H., Huang, J., & Rauss, P. J. (1998). The FERET database and evaluation procedure for face-recognition algorithms. *Image and vision computing*, *16*(5), 295-306.

[52] Benabderrahmane, S., Quiniou, R., & Guyet, T. (2014, August). Evaluating distance measures and times series clustering for temporal patterns retrieval. *Proceedings of the 15th {IEEE} International Conference on Information Reuse and Integration,* IRI 2014, Redwood City, CA, USA.

[53] Benabderrahmane, S., Smail-Tabbone, M., Poch, O., Napoli, A., & Devignes, M. (2010). *IntelliGO: A new vector-based semantic similarity measure including annotation origin. BMC Bioinformatics Journal, 11(588).*

[54] Benabderrahmane, S., Mellouli, N., & Lamolle, M. (2015). Learning temporal regression models and mapreduce voronoi tessellation for job offers recommendation. *Proceedings of International Conference on Data Mining,* Las Vegas, USA.



Mekami Hayet is a PhD candidate at the electrical engineering department of Djillali Liabes University, Algeria. She obtained her M.Sc. from the same university on 2004. Actually she is a teaching assistant in computer science department at Bel Abbes. Her research activities concern image processing, computer vision, robotics and face recognition problems. She attended a lot of international conferences and was involved in the organization of several workshops on images classification.

Sidahmed Benabderrahmane obtained his PhD of computer science from Nancy University, France in 2011 with a topic relative to data mining and classification problems. He earned his M.Sc on 2007 from Evry, Paris University. His research agenda includes the problems of big data analytics, machine learning, bioinformatics, computer vision, information retrieval, and knowledge extraction from complex data. He has several works which were presented at a lot of international conferences and journals. Actually he is a research associate at Paris 8 university, and teaching assistant at IUT de Montreuil, France.

Abdenacer Bounoua is a full professor of electrical engineering at Djillali Liabes University of Bel Abbes Algeria. He has a lot of research papers on image filtering, compression, and transmission. He has been teaching in the electrical engineering of Djillali Liabes University since 1985.

Abdelmalik Taleb Ahmed is a full professor at Valencienne University, France. He obtained his PhD from Lille 1 University, France in 1992. He then joined Valencienne University in 1999 as associate professor. He has a lot of conference and journal papers on image filtering, object detection and tracking, faces recognition.