Genetic Algorithm for Weight Optimization in Descriptor based Face Recognition Methods

Ladislav Lenc[1,2]
[1]*Dept. of Computer Science & Engineering, Faculty of Applied Sciences, University of West Bohemia, Plzeň, Czech Republic*
[2]*NTIS - New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Plzeň, Czech Republic*

Keywords: Face Recognition, Genetic Algorithm, Local Binary Patterns, LBP, Patterns of Oriented Edge Magnitudes, POEM.

Abstract: This paper presents a novel algorithm for weight optimization in descriptor based face recognition methods. We aim at the local texture features that are currently very popular in the face recognition (FR) field. Common concept in such methods is creating histograms of the operator values in rectangular image regions and concatenating them into one large vector called histogram sequence (HS). Usually the facial regions are given equal weight which does not correspond with the reality. We deal with this issue in this work and propose a novel method that optimizes the weights of the regions. The optimization method is based on a genetic algorithm (GA). We test the method together with the local binary patterns (LBP) and patterns of oriented edge magnitudes (POEM) descriptors. We evaluate our algorithms on two real-world corpora: Unconstrained facial images (UFI) database and FaceScrub database. The evaluation results show that the weighted methods outperform the non-weighted ones. The best achieved scores are 68.93% on the UFI database and 57.81% on the FaceScrub database.

1 INTRODUCTION

Face recognition (FR) methods based on local image features are recently very popular among the face recognition researchers. A significant advantage of these methods is a relatively easy computation and high recognition accuracy. These methods usually outperform holistic approaches such as Eigenfaces (Turk and Pentland, 1991) or Fisherfaces (Belhumeur et al., 1997).

There are a lot of various types of descriptors that were utilized for FR. Among others we can mention scale invariant feature transform (SIFT) (Lowe, 2004) or speeded-up robust features (SURF) (Bay et al., 2006). The SIFT and SURF methods proved very good results in many computer vision tasks but the use for FR is not as frequent. The majority of researchers recently concentrate on Gabor based features (Wiskott et al., 1997; Perez et al., 2011) and on the local texture features such as local binary patterns (LBP) (Ahonen et al., 2006). The newly designed descriptors aim at improved ability to capture important information contained in images. The matching is then performed using a simple nearest neighbour classifier or can employ a more sophisticated one such as support vector machines (SVM) or artificial neural networks (ANN).

The local texture features proved very good performance and therefore we chose this type of features for our experiments. We concentrate on the matching procedure of the image features created using the well known algorithms such as LBP or patterns of oriented edge magnitudes (POEM) (Vu et al., 2012). We would like to extend the concept of weighting that was proposed already in (Ahonen et al., 2006). In that work a simple procedure of assigning the weights is used. Image is split into rectangular regions of size 18×21 pixels The training set is then classified based on just one of the regions in that the image is partitioned. The weight is assigned according to the recognition score obtained using the particular window.

We believe that the weighting is very important and can leverage the performance of descriptor based methods. The main contribution of this work is thus the proposal of a novel weight optimization procedure based on a genetic algorithm (GA). Another important contribution is evaluation of the algorithm on two real-world corpora. We chose the UFI and Face-

Lenc, L.
Genetic Algorithm for Weight Optimization in Descriptor based Face Recognition Methods.
DOI: 10.5220/0005704403300336
In *Proceedings of the 8th International Conference on Agents and Artificial Intelligence (ICAART 2016) - Volume 2*, pages 330-336
ISBN: 978-989-758-172-4

Scrub (Ng and Winkler, 2014) databases. We evaluate the identification scenario which means we compare the test face against gallery faces and search for the most similar one. We use a scheme where the weights are trained on one database and tested on the other.

The rest of the paper is organized as follows. Section 2 summarizes important face recognition methods with a particular focus on the local image descriptors. Several methods employing genetic algorithms for optimization are also briefly described. Section 3 details the LBP and POEM algorithms, the comparison procedure and the proposed genetic algorithm for weight optimization. Section 4 reports the results of evaluation on the UFI and FaceScrub corpora. In section 5 we conclude the paper and give some possible directions for further research.

2 RELATED WORK

Face recognition (FR) is nowadays a well established field in computer vision and image analysis. The earlier methods were usually based on dimensionality reduction of image space using methods such as principal component analysis (PCA) (Turk and Pentland, 1991) or linear discriminant analysis (LDA) (Belhumeur et al., 1997). These methods use the face image as a whole and are usually called holistic. Other large group of methods that gets more attention recently is based on a set of image features.

LBP is one of the first local texture features utilized for face recognition. It was originally used for texture representation as presented in (Ojala et al., 1996). It is computed from the neighbourhood of a given pixel and uses the intensity of the central pixel as a threshold. The neighbouring pixels are compared to that threshold and are assigned either 0 or 1 (lower than threshold - 0, higher - 1). The binary digits are concatenated to a binary number and then converted to a decimal value. This value is used as a descriptor of the pixel.

The first application of LBP for face recognition was proposed by Ahonen et al. in (Ahonen et al., 2004). The face is divided into rectangular regions. In each region a histogram of LBP values is computed. All histograms are then concatenated into one vector called histogram sequence (HS) which is used for the face representation. The histogram intersection (HI) method or Chi square distance are used for vector comparison. A weighted LBP modification is also proposed in this work. It gives more importance to the regions around the eyes and the central part of the face. The reported recognition rate on the FERET dataset (Phillips et al., 1998) reaches 93% for the original method and 97% for the weighted LBP method.

One of the extensions of LBP are dynamic threshold local binary patterns (DTLBP) (Li et al., 2012). It takes into consideration the mean value of the neighbouring pixels and also the maximum contrast between the neighbouring points. According to the authors, this variation is less sensitive to noise

Another extension of the original method are Local Ternary Patterns (LTP) proposed in (Tan and Triggs, 2010). It uses three states to capture the differences between the center pixel and the neighbouring ones. Similarly to the DTLBP the LTP is less sensitive to the noise.

The so called Local Derivative Patterns (LDP) are proposed in (Zhang et al., 2010). In this case features of higher order are computed. This descriptor thus can represent more information than the original LBP.

The histogram for the original LBP contains 256 bins (8 bit binary number). However, it was proved that approximately 90% of patterns belongs to a subset of the values that are called uniform. The uniformity means that there are at most two transitions from 0 to 1 or from 1 to 0 in the binary number. The number of uniform patterns is 59 in case of 8-bit representation. The 59th bin is reserved for the non-uniform patterns. This idea brings significantly lower dimension of resulting features whereas the descriptive abilities are similar.

An interesting method utilizing uniform patterns is proposed in (Yang and Wang, 2007). The authors state that the histogram bin containing non-uniform patterns dominates among other bins and gives thus too much importance to this bin. Therefore they propose to assign such patterns to the closest uniform pattern (according to Hamming distance).

One of more recent descriptors is POEM. It was proposed in (Vu et al., 2012). It is based on the computation of gradients in the image pixels. The gradient orientations are then quantized and accumulated in a small region around the pixel called *cell*. The accumulated values are then compared with other cells within a larger region called *block*. The histograms are created for each direction in similar way as in LBP. This approach achieves very good results on FERET (97.6% for fb set) and LFW (75%) datasets.

Another interesting method proposed in (Vu, 2013) is called patterns of orientation difference (POD). Whereas POEM encodes the differences in edge magnitude distributions, POD encodes the distribution of orientations. It is assumed that this descriptor gives complementary information to POEM and the combination can improve the recognition results.

Local quantized patterns (LQP) (Hussain et al.,

2012) are a generalization of local features. It is based on vector quantization and thus reduces the number of patterns. Cosine similarity is used for face comparison. It is reported to achieve better results than other local feature methods. The accuracy on the FERET fb set is 99.9%.

A common property of the most of the above described LBP methods is that the images are divided into rectangular regions and histograms are computed in each region. All histograms from one image are concatenated and create the face representation. In the comparison stage each histogram can have different weight and thus can emphasize some important face parts. We would like to employ GA for weight optimization and therefore some GA approaches are described next.

GAs are a class of algorithms belonging to the group of evolutionary algorithms (EA). The idea was proposed by Holland in (Holland, 1973) and GAs are often used for optimization in many domains. It was also relatively early adopted in the image processing field. The key concept of GAs is the Darwin's theory of natural selection. The fittest specimens thus survive and create a better population in the next generation. Applying the rules of selection, crossover and mutation it goes through the search space with the goal to find the optimal solution.

One interesting application of GAs in computer vision is circle detection (Ayala-Ramirez et al., 2006). It detects circles in the edge image of a scene. Each candidate circle is described with its radius and coordinates of the center. Fitness function evaluates the real presence of the circle in the image. It is reported that this approach is capable of detecting circles with sub-pixel accuracy.

Another recent utilization of GA is color image segmentation (Abbasgholipour et al., 2011). It uses a permutation-coded genetic algorithm and aims at segmentation under varying lighting conditions.

An example use of GA for face recognition is proposed in (Al-Arashi et al., 2014). The GA was used for PCA performance optimization in this work.

A survey of GA based methods can be found in (Paulinas and Ušinskas, 2015).

3 PROPOSED METHOD

The method consists of three steps. The first one is feature extraction for which we employed the LBP and POEM operators. The second step is the weight optimization that determines the weights for the individual regions in the image. Final step is classification that compares the image representations and finds the most similar image from the gallery.

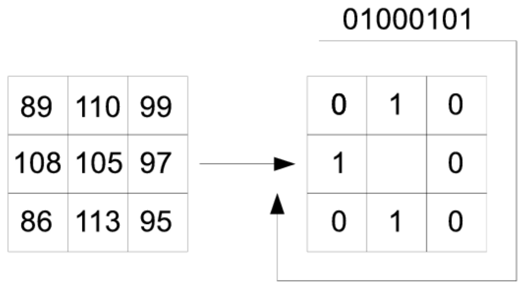

Figure 1: An example of the feature computing by the original LBP operator.

3.1 LBP Operator

The LBP operator was proposed in (Ojala et al., 1996). The original form uses a 3×3 neighbourhood of the central pixel. The algorithm assigns either 0 or 1 value to the 8 neighbouring pixels by Equation 1.

$$N = \begin{cases} 0 & \text{if } g_N < g_C \\ 1 & \text{if } g_N \geq g_C \end{cases} \tag{1}$$

where N is the binary value assigned to the neighbouring pixel, g_N denotes the gray-level value of the neighbouring pixel and g_C is the gray-level value of the central pixel. The resulting values are then concatenated into an 8 bit binary number. Its decimal representation is used for further computation. This approach is illustrated in Figure 1.

The original LBP operator was further extended to use circular neighbourhoods of various sizes and also with different numbers of points. A bilinear interpolation is used to compute the values in the points that are not placed in the pixel centres. The LBP is then denoted as $LBP_{P,R}$ where P is the number of points and R is the radius of the neighbourhood. The value of the operator is computed by Equation 2.

$$LBP_{P,R} = \sum_{p=1}^{P-1} s(g_p - g_c)2^p, S(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases} \tag{2}$$

where g_p denotes the points on the circle and g_c is the central point. Figure 2 illustrates the computation of $LBP_{8,2}$ operator.

3.2 POEM Operator

The POEM descriptor was proposed in (Vu et al., 2012). First the gradient in each pixel of the input image is computed. An approximation utilizing simple convolution operator such as Sobel or Scharr is used to compute gradients in the x and y directions. These values are used for computation of gradient orientation and magnitude.

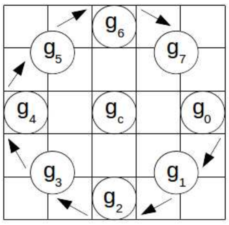

Figure 2: Computation of $LBP_{8,2}$ operator.

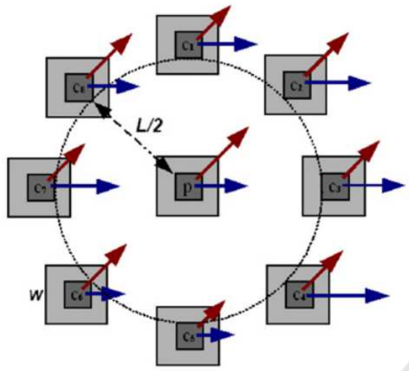

Figure 3: Depiction of POEM computation (Vu et al., 2012). Light gray square regions around pixels represent the *cells* and larger surroundings with diameter L is called *block*. Arrows represent the accumulated gradients.

The gradient orientations are then discretized. The number of orientations is denoted d and is usually set to 3. Each pixel is now represented as a vector of length d. It is a histogram of gradient values in a small square neighbourhood of a given pixel called *cell*. Figure 3 depicts the meaning of *cell* and *block* terms.

The final encoding similar to that of LBP is done in a round neighbourhood with diameter L called *block*. The 8 cell values are compared with the central one and the binary representation is created. It is computed for each gradient orientation and thus the descriptor is d times longer than in case of LBP.

3.3 Feature Vector Creation and Comparison

The facial image is first divided into N non-overlapping rectangular regions. Histogram of operator values is computed in each region. The dimension of the histogram depends on the operator used. The individual histograms are finally concatenated into one large vector called histogram sequence. We compare the representation using weighted χ^2 distance.

$$d(S,T) = \sum_{i=1}^{N} w_i \sum_{j=1}^{L} \frac{(S_i(j) - T_i(j))^2}{2(S_i(j) + T_i(j))} \tag{3}$$

where S and T are two histogram sequences composed from N histograms, w_i is the weight assigned to each histogram and L is the dimension of one histogram.

3.4 Weights Optimization

The optimal setting of weights of individual image regions is the key contribution of this work. We employ a genetic algorithm for this task. It is based on differential evolution.

We adopted the three main concepts commonly used in GAs: selection, crossover and mutation. One specimen is represented as a vector of weights for the image regions. The length of the vector depends on the size of the regions.

We begin with a randomly generated population that is iteratively evolved and evaluated according to the fitness function given by equation 4.

$$C\frac{hits}{hits + misses} + \frac{nc}{c} \tag{4}$$

where C is a large constant determined experimentally and hits and misses are the correctly and incorrectly classified examples respectively. nc is a value that accumulates similarities of faces from different classes. c is the accumulated value of similarities of faces belonging to the same class.

4 EXPERIMENTAL SETUP

This section describes the corpora that we used for evaluation and the results obtained on these datasets.

4.1 Corpora

We concentrate on challenging real-world corpora that are recently used.

4.1.1 FaceScrub

This large database of real-world images was collected using an approach described in (Ng and Winkler, 2014). It contains images of 530 individuals equally split to males and females. The total number of images is 107,818 with an average of 203 images per person. The database is distributed as a list of URLs where the images are stored. Each line in the list contains the URL, name, coordinates of the facial region and a checksum. The images thus have to be downloaded and cropped according to the provided information.

Figure 4: Example images from the FaceScrub database.

We downloaded and cropped the faces automatically. Only the images with correct checksum were downloaded. Some of the URLs were also not available in the moment of downloading. The resulting dataset thus contains the 530 individuals but the total number of images is somewhat lower. It is 80,167 which gives an average of 151 images per person.

We didn't find any closer specification of testing protocol together with this dataset. Therefore, we divided the database into two parts. Th first one is used as training set and the other one as test set. The division was performed randomly. Figure 4 shows three example images from this dataset.

4.1.2 Unconstrained Facial Images

The UFI database is composed of face images taken in real-world conditions and is freely available for research purposes at `http://ufi.kiv.zcu.cz`. It is composed of two partitions. The first one is called *Cropped images* and contains automatically detected faces from photographs. The number of individuals is 605. These images are cropped and resized to have approximately the same face size.

The second partition called *Large images* contains larger image regions that contain not only the face. Some background is also present and the face may be located anywhere in the image. Therefore, it is necessary to include the face detection task before the face recognition itself. This partition contains images of 530 individuals.

We used the *Cropped images* partition in our experiments. The distribution of number of examples is showed in figure 5.

We used the testing protocol designed by the authors of the database. Figure 6 shows example images from the UFI database.

4.2 Results

We evaluated our method on the two above described corpora. The weight masks are always learned on the other corpus to achieve unbiased results.

First, we tested the UFI database. To create the masks, we used a portion of the FaceScrub database. We chose just 5 images of one individual for the training set and other 5 for test set. We experimented with

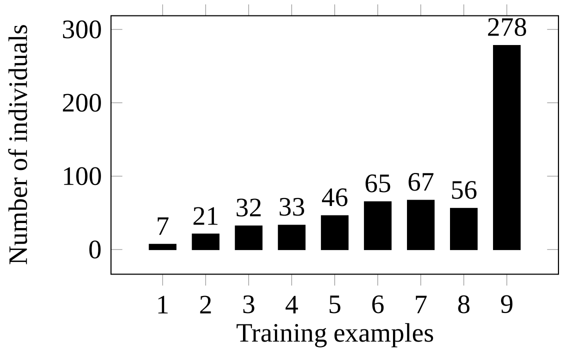

Figure 5: Distribution of the training image numbers in the *Cropped images* partition.

Figure 6: Example images from the UFI database.

3 different numbers of individuals (20, 50 and 100). The resulting masks for LBP operator are depicted in figure 7 while the masks for POEM operator are shown in figure 8. The white parts have high weights whereas the black ones low.

The obtained accuracies are reported in table 1. The upper part shows results when the LBP operator is used while the lower part gives the results for POEM operator. The first line in each part gives the accuracy obtained with the basic non-weighted variant of the method. The best results are in bold face.

The best score for LBP is obtained with the mask trained on 100 individuals and reaches 54.71%. In the case of POEM, the best accuracy 68.93% is achieved with mask trained on 50 individuals.

The recognition accuracy for the FaceScrub database is measured in a similar manner. We trained the weight masks on different subsets of the UFI database. We chose 50, 100 and all (605) individu-

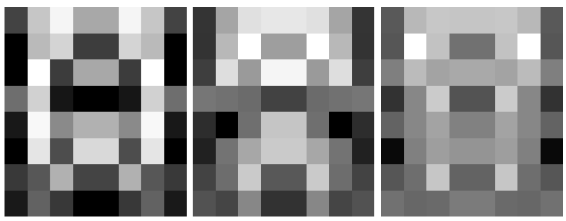

Figure 7: Weight masks trained on the FaceScrub database and LBP operator with 20, 50 and 100 individuals (from left to right).

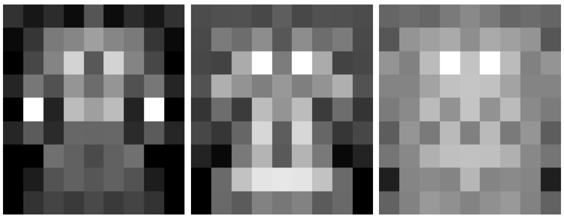

Figure 8: Weight masks trained on the FaceScrub database and POEM operator with 20, 50 and 100 individuals (from left to right).

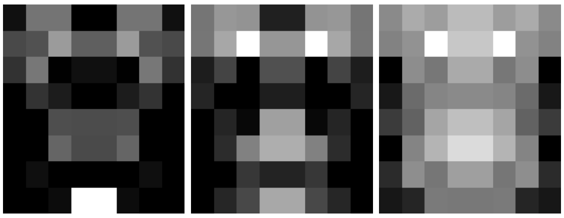

Figure 9: Weight masks trained on the UFI database and LBP operator with 50, 100 and all individuals (from left to right).

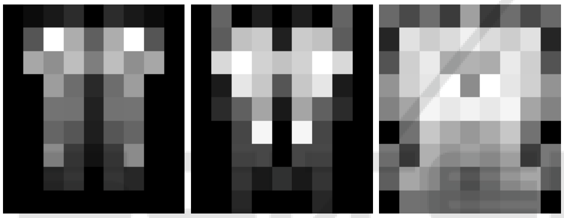

Figure 10: Weight masks trained on the FaceScrub database and POEM operator with 50, 100 and all individuals (from left to right).

als. The learned masks for LBP and POEM operators are depicted in figure 9 and figure 10 respectively.

Table 2 shows the accuracies obtained on the FaceScrub database. The upper part shows results for LBP and the lower one for POEM. The best results are in bold face.

The highest accuracy for LBP operator is 47.52% when the mask is trained on the whole UFI database. The best result for POEM is 57.81% also with the mask trained on all individuals from UFI. The results indicate that the test scheme we used for FaceScrub database is even more challenging than the one used for UFI database.

5 CONCLUSIONS AND PERSPECTIVES

In this paper we proposed a novel method for weight optimization in the descriptor based face recognition methods. It is based on a genetic algorithm and tries to determine optimal weights on a set of facial images We used the optimized weights together with the LBP and POEM descriptors. The method is tested on two challenging real-world corpora, namely FaceScrub and UFI. We show that the results are promising and outperform the basic non-weighted methods. The accuracy on the UFI database shows improvement of 5.4% with the LBP operator and 1.7% with the POEM operator compared to the non-weighted methods. In the case of FaceScrub database the improvement is 3.2% with LBP and 1.5% with POEM.

The proposed method have many possibilities for further improvement. One of them is to improve the initial conditions of the GA. The start weights could be possibly assigned according to a Gaussian distribution which would emphasize the central parts of the face. Also the fitness function can be probably improved.

Table 1: Recognition results on the UFI database.

	Recognition rate [%]
LBP non-weigted	49.26
LBP mask-20	52.73
LBP mask-50	54.55
LBP mask-100	**54.71**
POEM non-weighted	65.95
POEM mask-20	68.43
POEM mask-50	**68.93**
POEM mask-100	67.60

Table 2: Recognition results on the FaceScrub database.

	Recognition rate [%]
LBP non-weighted	44.27
LBP mask-50	42.34
LBP mask-100	43.66
LBP mask-all	**47.52**
POEM non-weighted	56.35
POEM mask-50	56.60
POEM mask-100	56.60
POEM mask-all	**57.81**

ACKNOWLEDGEMENTS

This work has been supported by the project LO1506 of the Czech Ministry of Education, Youth and Sports. Special thanks belong to Vojtěch Košař for implementation and valuable ideas. We also would like to thank Czech New Agency (ČTK) for support and for providing the photographic data.

REFERENCES

Abbasgholipour, M., Omid, M., Keyhani, A., and Mohtasebi, S. (2011). Color image segmentation with genetic algorithm in a raisin sorting system based on machine vision in variable conditions. *Expert Systems with Applications*, 38(4):3671–3678.

Ahonen, T., Hadid, A., and Pietikäinen, M. (2004). Face recognition with local binary patterns. In *Computer vision-eccv 2004*, pages 469–481. Springer.

Ahonen, T., Hadid, A., and Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):2037–2041.

Al-Arashi, W. H., Ibrahim, H., and Suandi, S. A. (2014). Optimizing principal component analysis performance for face recognition using genetic algorithm. *Neurocomputing*, 128:415–420.

Ayala-Ramirez, V., Garcia-Capulin, C. H., Perez-Garcia, A., and Sanchez-Yanez, R. E. (2006). Circle detection on images using genetic algorithms. *Pattern Recognition Letters*, 27(6):652–657.

Bay, H., Tuytelaars, T., and Van Gool, L. (2006). Surf: Speeded up robust features. In *Computer vision–ECCV 2006*, pages 404–417. Springer.

Belhumeur, P. N., Hespanha, J. P., and Kriegman, D. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):711–720.

Holland, J. H. (1973). Genetic algorithms and the optimal allocation of trials. *SIAM Journal on Computing*, 2(2):88–105.

Hussain, S. U., Napoléon, T., and Jurie, F. (2012). Face recognition using local quantized patterns. In *British Machive Vision Conference*, pages 11–pages.

Li, W., Fu, P., and Zhou, L. (2012). Face recognition method based on dynamic threshold local binary pattern. In *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service*, pages 20–24. ACM.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

Ng, H.-W. and Winkler, S. (2014). A data-driven approach to cleaning large face datasets. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 343–347. IEEE.

Ojala, T., Pietikäinen, M., and Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59.

Paulinas, M. and Ušinskas, A. (2015). A survey of genetic algorithms applications for image enhancement and segmentation. *Information Technology and control*, 36(3).

Perez, C. A., Cament, L. A., and Castillo, L. E. (2011). Methodological improvement on local gabor face recognition based on feature selection and enhanced borda count. *Pattern Recognition*, 44(4):951–963.

Phillips, P. J., Wechsler, H., Huang, J., and Rauss, P. (1998). The FERET database and evaluation procedure for face recognition algorithms. *Image and Vision Computing*, 16(5):295–306.

Tan, X. and Triggs, B. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *Image Processing, IEEE Transactions on*, 19(6):1635–1650.

Turk, M. A. and Pentland, A. P. (1991). Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE.

Vu, N.-S. (2013). Exploring patterns of gradient orientations and magnitudes for face recognition. *Information Forensics and Security, IEEE Transactions on*, 8(2):295–304.

Vu, N.-S., Dee, H. M., and Caplier, A. (2012). Face recognition using the poem descriptor. *Pattern Recognition*, 45(7):2478–2488.

Wiskott, L., Fellous, J.-M., Kuiger, N., and Von Der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):775–779.

Yang, H. and Wang, Y. (2007). A lbp-based face recognition method with hamming distance constraint. In *Image and Graphics, 2007. ICIG 2007. Fourth International Conference on*, pages 645–649. IEEE.

Zhang, B., Gao, Y., Zhao, S., and Liu, J. (2010). Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. *Image Processing, IEEE Transactions on*, 19(2):533–544.