Ensemble of Patterns of Oriented Edge Magnitudes Descriptors For Face Recognition

Loris Nanni,[1] Alessandra Lumini,[2] Sheryl Brahnam,[3] Mauro Migliardi[1]

[1]DEI, University of Padua, viale Gradenigo 6, Padua, Italy. {loris.nanni, mauro.migliardi}@unipd.it;
[2]DEI, Università di Bologna, Via Venezia 52, 47521 Cesena, Italy. alessandra.lumini@unibo.it;
[3]Computer Information Systems, Missouri State University, 901 S. National, Springfield, MO 65804, USA. sbrahnam@missouristate.edu

Abstract

In this work we propose an ensemble of descriptors for face recognition. Starting from the base patterns of the oriented edge magnitudes (POEM) descriptor, we developed different ensembles by varying the preprocessing techniques, the parameters for extracting the accumulated magnitude images (AM), and the parameters of the local binary patterns (LBP) applied to AM. Our best proposed ensemble works well regardless of whether dimensionality reduction by principal component analysis (PCA) is performed or not before the matching step. We validate our approach using the FERET datasets and the Labeled Faces in the Wild (LFW) dataset. We obtain very high performance rates in both datasets. To the best of our knowledge, we obtain one of the highest performances reported in the literature on the FERET datasets. We want to stress that our ensemble obtains these results without combining different texture descriptors and without any supervised approach or transform. Finally, two cloud use cases are proposed.
The MATLAB source of our best approach will be freely available: http://www.dei.unipd.it/wdyn/?IDsezione=3314&IDgruppo_pass=124

Keywords: Face recognition; ensemble of descriptors; patterns of oriented edge magnitudes; local binary patterns.

1 Introduction

The problem of face recognition has been considered since the very beginning of computer vision. In the last two decades it has been extensively studied due to the large number of government and commercial applications that require the development of robust and reliable systems. In general, there are three main categories of applications for face recognition: 1) face verification, which authenticates a person's identity by comparing his or her face with a corresponding template; 2) face identification, which recognizes a person's face by comparing it with a dataset of faces; and 3) face tagging, which is a particular case of face identification.

Different techniques have been proposed in the literature for face recognition, including Principal Component Analysis (PCA), Discriminant Analysis, Local Binary Patterns (LBP), neural networks, Elastic Template Matching, Algebraic moments, and many other ad hoc descriptors and classifiers. Existing face recognition techniques have be classified into four types [1], based on the way they define the face: 1) *appearance based approaches*, which use global texture features (including Eigenfaces [2] and other linear transformation approaches); 2) *model based approaches*, which work on the shape and the texture of the face, along with 3D depth information; 3) *geometry or template based approaches*, which compare the input image with a set of templates constructed either by using statistical tools or by analyzing local facial features and their geometric relationships (including Elastic Bunch Graph Matching algorithms [3]); and 4) *techniques using Neural Networks*, which are often used in combination with Gabor Filters [4].

For many applications, face recognition performance has reached a satisfactory level under the frontal pose and optimal lighting conditions. Performance degrades, however, with pose and lighting variations and in uncontrolled environments.

To deal with these problems, researches have focused their studies on the design of robust face descriptors that are not only discriminative but also insensitive to pose variations, changes in facial expression, and lighting conditions. For example, Pinto et al. [5] use V1-like and Gabor filters for face representation. Cao el al. [6] propose a method to encode the local micro-structures of a face into a set of more uniformly distributed discrete codes. In [7] and [8] a novel descriptor, called the Patterns of Oriented Edge Magnitudes (POEM) is proposed. POEM is an oriented spatial multi-resolution descriptor that captures rich information (self-similarity structure) about the original image. Other encouraging results in difficult conditions have been obtained in [9] using a sparse representation to select a feature for person-specific verification. The last two works aim to solve one of the most difficult problems in face recognition: descriptors gaining high recognition performance are usually computationally intensive, while low-complexity methods often do not perform reliably enough.

We start our study from one of the most efficient and high performing descriptors recently proposed in the literature: the POEM [7]. In this work we try to boost the performance of POEM, not by combining it with other texture descriptors (as in [8]), but by building an ensemble based on the variation of its parameters and of the enhancement approaches used before the feature extraction step. The most interesting finding is that building an ensemble in this easy way boosts the performance obtained by POEM. Another contribution of this paper is the definition of some variants of the POEM descriptor using dense LBP [10], instead of LBP, for representing the AM images and for filtering the enhanced image by Gabor filters [11] before the POEM extraction step.

2 Ensemble of POEM Descriptors

2.1 The POEM Descriptor

The poem descriptor is based on the idea of characterizing the local face appearance and shape by the distribution of local intensity gradients, or edge directions.

The POEM feature extraction consists of three steps (see [12] for mathematical details):

1. Gradient computation and orientation quantization: first the gradient image is computed, then orientation of each pixel is discretized over 0-π (unsigned representation) or 0-2π (signed representation) (we use the unsigned representation).
2. Magnitude accumulation: a local histogram of orientations over all pixels within a local image patch (cell) is calculated to incorporate information from neighboring pixels.
3. Self-similarity calculation: the accumulated magnitudes are encoded across different directions using the self-similarity LBP-based operator within a larger patch (block). The final POEM descriptor at each pixel is the concatenation of all unidirectional POEMs at different orientations.

2.2 Designing an Ensemble of POEM Variants

In this work, we build our ensemble as follows:

- We propose to enhance the image using different methods, for each method a different POEM descriptor is extracted and used to train a classifier; the preprocessing techniques used in this work are detailed in section 2.3;
- The POEM descriptor depends on a high number of parameters that should be fine-tuned to the application: a) the number of orientations discretized, b) the size of the cell, c) the size of the block, and d) the number of neighbors considered in LBP. Instead of using a single set of optimized parameters, several descriptors are extracted using different sets of parameters, as reported in table 1. These sets of parameters have been proven to work quite well in several different datasets without ad hoc optimization (see [12]).

Number of orientations	cell size	Block size	LBP Neighbors
3	7	5	8
4	7	5	8
3	7	6	8
7	7	5	8
3	4	5	8
3	6	5	8
3	8	5	8
3	7	5	6
3	7	5	9
3	7	5	12

Table 1. Set of parameters used for building an ensemble

2.3 Prepressing Techniques

Image enhancement has advanced greatly in face recognition, especially when dealing with the problem of illumination changes. In this work we examined the following approaches:

- Adaptive single scale retinex (AR): The adaptive single scale retinex algorithm [13] is a variant of the retinex technique, which aims at improving poor scene detail and color reproduction in dark areas of the image. This method gained the best performance in our experiments;
- Anisotropic smoothing (AS): Introduced by Gross and Brajovic in [14], the algorithm computes the estimate of the illumination field and then compensates for it according to some aspects of human visual perception with the aim of enhancing the local contrast of the image;
- Difference of Gaussians (DG): This is a filtering-based normalization technique that relies on the difference of a Gaussians filter to produce a normalized image. This is accomplished by applying a bandpass filter to the input image (note: the log transform is applied to the image [15] before the filter is used);
- Gabor filtering (GF): The last preprocessing method used in this work is not an enhancement technique *per se* but rather a filter. Before the feature extraction step, the input image is filtered by a bank of Gabor filters (using the same Gabor's settings as in [3]).

3 Experimental Results

3.1 Datasets

We test our proposed ensembles using the FERET [11] and LFW [13] benchmark databases. The FERET database images are divided into five datasets: Fa, Fb, Fc, Dup1, and Dup2. Fa is the training set, and the other sets are used for testing. Fb contains pictures taken on the same day as the Fa pictures and with the same camera and illumination conditions. Fc contains pictures taken on the same day as the Fa pictures, but with different cameras and with different illumination conditions. Dup1 and Dup2 contain pictures taken on different days than the Fa pictures were taken, but within a year for Dup1 and longer than one year for Dup2. In our experiments, the FERET gray images are aligned using the true eyes position and cropped to 110 × 110 pixels.

The LFW [13] database contains 13233 images of celebrities. It is very challenging since it includes great variations in terms of lighting, pose, age, and even image quality. Two views of the database are provided. View 1, which is used for model selection only, contains a training set of 2200 face pairs and a testing set of 1000 face pairs. View 2 is for performance reporting, and is made up of 10 non-overlapping sets of 600 matches that can be used for 10-fold cross-validation of algorithms and parameters developed on View 1. In our experiments, the LFW gray images are aligned automatically according to the procedure described in [8] and cropped to 110 × 110 pixels.

3.2 Results

We test our ensembles on both databases using their official testing protocols. The performance indicator is the accuracy for the problem of person identification using the FERET dataset. For the LFW dataset, the classification accuracy of each match between two faces is either genuine or impostor (see [13] for more details). In Table 2, we provide a detailed description of the methods compared in our experiments, according to the following parameters:

- Preprocessing procedure: no preprocessing (NO), Adaptive single scale retinex (AR), Anisotropic smoothing (AS), Difference of Gaussians (DG), Gabor filtering (GF). Gabor filtering is applied both to original (NO) or enhanced image, according to the settings used in [3] (4 scales and 4 directions);
- Self-similarity calculation (SSC): LPB or Dense LBP (DLBP) [10] are used for the self-similarity calculation step of POEM;
- Dimensionality reduction and distance measure (DD): city block distance (CBD) is used to compare high dimensional POEM descriptors (in the original code, the chi-square distance (CS) is used), while angle distance (AD) is used when the descriptor is reduced to a lower space by PCA. In this work, we vary from [14] by using the same dimensionality parameter (D=500) and the same projection space (trained on FERET training set) for all the experiments (for both FERET and LFW). Moreover, when PCA is applied the square root normalization is performed before the matching, as in [14];
- Stand-alone/ensemble (SE): ensemble approaches are obtained by perturbing POEM parameters (see Section 2.2) or by perturbing the preprocessing techniques (see Section 2.3). The scores are fused by sum rule. We define: SA, stand-alone method; Ep, the perturbation of POEM parameters; Ee, perturbation of the preprocessing techniques; and E, the perturbation of both preprocessing techniques and the POEM parameters.

In Table 3, we report the accuracy obtained by our approaches on both databases. It should be noted that in order to test the robustness of our approach the same PCA projection matrix calculated in the FERET training set is used in LFW. Moreover, due to computational issues, only a subset of the proposed approaches (the most interesting ones) are tested on LFW.

By examining Table 3, the following conclusions can be drawn:

- Our ensembles are similar in performance to each other, and they outperform [14] in the FERET dataset without any strong optimization (i.e., by using the same parameter settings for the four FERET datasets and the LFW dataset);
- In LFW, the authors of [14] claim the highest results using an ad hoc projection matrix. In contrast, we use the projection matrix for PCA that is constructed using training images of the FERET dataset;
- It is clear that our idea for designing classifiers boosts the performance of the base POEM descriptor in both datasets (please note that $POEM^{PCA}_{LBP}(ar)$ is based on the original code shared by [14]);
- Dense LBP obtains the same performance when the PCA projection is performed but outperforms LBP when the projection is not performed;
- Almost all the proposed ensembles outperform the stand-alone versions;
- Due to lack of space, we report comparisons only with already proposed POEM systems {Vu, 2012 #3994} [14]; it is clear that our system obtains good results without tuning the system for a given dataset (the same approach is used for both datasets). In [8] [12] [14] several state-of-the-arts approaches are compared. Our system obtains performance similar to the best approach tested in the FERET dataset (only two methods outperform our system: they obtain an average accuracy of 96.9% and 97.7% in the FERET dataset);
- The proposed system works well on the LFW dataset but not as well as other approaches. However, POEM has significantly lower complexity with respect other competing systems, which offsets this performance difference to some degree.

Name	Preprocessing	SSC	DD	SE	Description
$POEM_{LBP}^{CS}$	-	LBP	CS	SA	POEM descriptor (source code from [12])
$POEM_{LBP}$	-	LBP	CBD	SA	The method above using CBD
$POEM_{LBP}^{PCA}$	-	LBP	PCA+AD	SA	code of [14] with fixed PCA dimension (500)
$POEM_{LBP}^{PCA}(ar$	AR	LBP	PCA+AD	SA	The method above using preprocessing
$POEM_{LBP}(ar$	AR	LBP	CBD	SA	The method above without PCA
$POEM_{DLBP}^{PCA}$	-	DLBP	PCA+AD	SA	Use of DenseLBP
$E\_P_{LBP}^{PCA}(ar)$	AR	LBP	PCA+AD	Ep	Perturbation of POEM parameters
$E\_P_{LBP}(ar)$	AR	LBP	CBD	Ep	Perturbation of POEM parameters
$E\_P_{LBP}^{PCA}(\cdot)$	(AR, AS, DG)	LBP	PCA+AD	Ee	Perturbation of enhancement
$E\_P_{LBP}(\cdot)$	(AR, AS, DG)	LBP	CBD	Ee	Perturbation of enhancement
$E\_P_{LBP}^{PCA}(e)$	(AR, AS, DG)	LBP	PCA+AD	E	Perturbation of enhancement and parameters
$E\_P_{LBP}(e)$	(AR, AS, DG)	LBP	CBD	E	Perturbation of enhancement and parameters
$E\_P_{DLBP}^{PCA}(\cdot)$	(AR, AS, DG)	DLBP	PCA+AD	Ee	Perturbation of enhancement
$E\_P_{DLBP}(\cdot)$	(AR, AS, DG)	DLBP	CBD	Ee	Perturbation of enhancement
$E\_P_{DLBP}^{PCA}(gf)$	(AR, AS, DG)+GF	DLBP	PCA+AD	Ee	Perturbation of enhancement and of GF
$E\_P_{DLBP}(gf)$	(AR, AS, DG)+GF	DLBP	CBD	Ee	Perturbation of enhancement and of GF

Table 2. Compared approaches.

	FERET Datasets					LFW Dataset
Method	**Fb**	**Fc**	**Dup1**	**Dup2**	*Average*	
$POEM_{LBP}^{CS}$	95.2	95.9	77.1	77.4	86.4	74.3
$POEM_{LBP}$	95.7	96.4	77.0	79.5	87.1	74.3
$POEM_{LBP}^{PCA}$	98.5	97.9	87.8	82.9	91.7	-
$POEM_{LBP}^{PCA}(ar)$	98.5	**100**	90.4	89.3	94.5	74.9
$POEM_{LBP}(ar)$	94.1	98.5	77.3	78.6	87.1	-
$E\_P_{LBP}^{PCA}(ar)$	98.6	**100**	91.3	90.6	95.1	75.2
$E\_P_{LBP}(ar)$	94.4	98.0	78.4	79.1	87.4	-
$E\_P_{LBP}^{PCA}(\cdot)$	98.7	**100**	94.6	93.6	**96.7**	**76.9**
$E\_P_{LBP}(\cdot)$	95.2	99.0	81.9	82.5	89.6	-
$E\_P_{LBP}^{PCA}(e)$	98.9	**100**	94.3	93.6	**96.7**	76.8
$E\_P_{LBP}(e)$	95.2	99.0	81.4	81.2	89.2	-
$POEM_{DLBP}^{PCA}$	98.7	99.0	88.1	83.8	92.4	-
$E\_P_{DLBP}^{PCA}(\cdot)$	**98.8**	**100**	94.2	**94.0**	96.7	76.6
$E\_P_{DLBP}(\cdot)$	95.1	99.5	84.2	85.5	91.0	-
$E\_P_{DLBP}^{PCA}(gf)$	98.7	**100**	**94.7**	93.6	**96.7**	-
$E\_P_{DLBP}(gf)$	95.1	99.5	83.8	85.0	90.8	-
POEM [12]	98.1	99.0	79.6	79.1	88.9	75.4
POEM+PCA [14]	99.6	99.5	88.8	85.0	93.2	82.7

Table 3. Accuracy obtained by the methods proposed in this paper in FERET and LFW databases.

4 Two Use Cases

We now present two use cases based on smartphone-plus-cloud infrastructure. In the first case, we consider a face tagging service available to private smartphone users. The recent diffusion of smartphones has provided an unbounded source of photos that continues to grow daily. Many of these pictures end up inside social networks where they might automatically be tagged by the social network system. What we propose is a mechanism for automated face tagging where the results are delivered directly to the smartphone for user filtering before social sharing. This mechanism could be further enhanced to allow automated sharing of specific photos with selectable sets of friends tagged. In figure 1 we show the general architecture of the system.

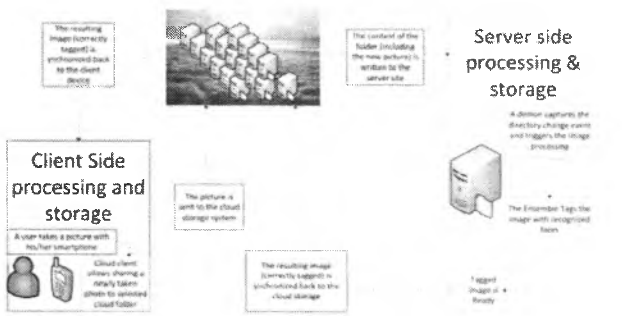

Figure 1 The general architecture of the system.

Several cloud storage providers have developed smartphone apps that allow both the direct uploading of photos taken by devices to personal cloud storage and the sharing of photos with other users. We suggest configuring the system so that when a photo is shared with a service provider the user would automatically trigger the processing of the photo by the server side face tagging application. To be more specific, each user would share a folder in the cloud with the face tagging service provider. A photo that needed to be processed would be uploaded to that shared folder using the cloud application, and then it would be synchronized to the server side folder by the cloud system. The appearance of this new photo would trigger the face tagging application that would in turn process it and generate results. These results would be written in the same directory so that the cloud application could automatically store it back into the cloud and synchronize it back to the smartphone. The mechanism for synchronization back to the smartphone might be provided by the app itself, or it might require the development of a special add-on leveraging the cloud API.

In our proof of concept experiments, we have tested an android app for the commercial cloud storage provider Syncplicity.[1] This app provides push notifications and automated sync-back for selected folders. Other solutions, such as Dropbox,[2] could be used, but they fail to provide automated sync-back to the smartphone. However, they have APIs that allow developing dedicated add-ons for such tasks.

Both solutions described above are based on cloud servers that are out of the control of both the smartphone user and the face tagging service provider since they are based on commercial cloud storage solutions. It is also possible to adopt a different approach leveraging the open source software provided by the OwnCloud project.[3] This project is dedicated to the development of an open source cloud server and clients for several desktop operating systems, such as Microsoft Windows, Linux, MacOS, and mobile operating systems, such as iOS and Android.

By leveraging this software, it would be possible for a face-tagging service provider to control the cloud storage. The absence of a third party storing the data would be especially relevant in the scenario of a security system. As an example, consider our second use case of a video surveillance system. Security cameras would scan choke-points in the area to be monitored in order to get clearer pictures of the people present. Both for privacy and for security reasons, it would not be possible to store these pictures on a public cloud; they would have to be sent to a private cloud hosted inside the premises of the face recognition server. In order to provide this level of security, an open source cloud solution, such as the above mentioned OwnCloud, could be used. Once the photographs have reached the face recognition server, they would be compared to a database of known persons of interest. In the case of a positive match, the system would push the picture and, possibly, a text file with a brief description of the subject to the smartphones of all the security agents on the premises.

5 Conclusion

In this paper, we have shown that it is possible to improve the performance of a single descriptor (POEM) by building an ensemble obtained by perturbing some steps in the face recognition process. In particular, our experiments show that the most reliable approach for building an ensemble is to perturb the enhancement method.

The main novelties our proposed system are the following: 1) our experiments show that it is possible to improve considerably a stand-alone descriptor by changing its parameters; 2) we also show that another easy way to boost the performance of a pattern recognition system is to use different enhancement techniques, and 3) some variants of the base POEM are proposed (e.g., using different

[1] EMC, Syncplicity, www.syncplicity.com, last retrieved on February the 13th 2013

[2] The Dropbox tour, www.dropbox.com/tour, last retrieved on February the 13th 2013

[3] OwnCloud, owncloud.org, last retrieved on February the 13th 2013

descriptors applied to AM or to filter the image by Gabor filters before the AM extraction) and are shown to enhance performance. Finally, two cloud use cases are outlined.

The main drawback of the proposed system is the increase computation time with respect to stand-alone methods. For example, considering $E\_P_{LBP}^{PCA}(\cdot)$, the time for the enhancement and the feature extraction processes is ~1 second, while the matching time is ~ 0.00013 seconds (Intel i5 - 3.3GhZ - 8GRAM - parallelized Matlab code).

Suggestions for future experiments would include a) testing other feature transformations before the matching step, b) combining our proposed POEM-based approach with other descriptors, and c) testing other texture descriptors, instead of LBP, for representing the AM images of POEM.

References

[1] Muruganantham, S., and Jebarajan, T., "A comprehensive review of significant researches on face recognition based on various conditions," *International Journal of Computer Theory and Engineering,* vol. 4, no. 1, pp. 7-15, 2012.

[2] Turk, M. A., and Pentland, A. P., "Eigenfaces for recognition," *Journal of Cognitive Neuroscience,* vol. 3, no. 1, pp. 71-86, 1991.

[3] Zhang, B., Shan, S., Chen, X., and Gao, W., "Histogram of gabor phase patterns (HGPP): A novel object representation approach for face recognition," *IEEE Transactions of Image Processing,* vol. 16, no. 1, pp. 57-68, 2007.

[4] Bhuiyan, A.-A., and Liu, C. H., "On face recognition using gabor filters," in World Academy of Science, Engineering and Technology, 2007, pp. 51-56.

[5] Pinto, N., DiCarlo, J., and Cox, D., "How far can you get with a modern face recognition test set using only simple features?," in CVPR'09, 2009.

[6] Cao, Z., Yin, Q., Tang, X., and Sun, J., "Face recognition with learning-based descriptor," pp. 2707-2714, 2010.

[7] Vu, N.-S., and Caplier, A., "Face recognition with patterns of oriented edge magnitudes," in ECCV, 2010.

[8] Vu, N.-S., "Exploring patterns of gradient orientations and magnitudes for face recognition," *IEEE Transactions on Information Forensics and Security,* vol. 8, no. 2, pp. 295-304 2013.

[9] Liang, Y., Liao, S., Wang, L., and Zou, B., "Exploring regularized feature selection for person specific face verification," in ICCV, 2011.

[10] Ylioinas, J., Hadid, A., Guo, Y., and Pietikäinen, M., "Efficient image appearance description using dense sampling based local binary patterns," in Asian Conference on Computer Vision, 2012.

[11] Phillips, J., Moon, H., Rizvi, S. A., and Rauss, P. J., "The feret evaluation methodology for face-recognition algorithms," *IEEE Transactions of Pattern Analysis and Machine Intelligence,* vol. 22, pp. 1090-1104, 2000.

[12] Vu, N.-S., Dee, H. M., and Caplier, A., "Face recognition using the POEM descriptor," *Pattern Recognition Letters,* vol. 45, no. 7, pp. 2478-2488, 2012.

[13] Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E., *Labeled faces in the wild: a database for studying face recognition in unconstrained environments,* vol. Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[14] Vu, N.-S., and Caplier, A., "Enhanced patterns of oriented edge magnitudes for face recognition and image matching," *IEEE Transactions on Image Processing,* vol. 21, no. 3, pp. 1352-1365, 2012.