Learning Discriminative Local Binary Patterns for Face Recognition

Daniel Maturana, Domingo Mery and Alvaro Soto
{dimatura,dmery,asoto}@puc.cl
Department of Computer Science, Pontificia Universidad Católica de Chile

***Abstract*— Histograms of Local Binary Patterns (LBPs) and variations thereof are a popular local visual descriptor for face recognition. So far, most variations of LBP are designed by hand or are learned with non-supervised methods. In this work we propose a simple method to learn discriminative LBPs in a supervised manner. The method represents an LBP-like descriptor as a set of pixel comparisons within a neighborhood and heuristically seeks for a set of pixel comparisons so as to maximize a Fisher separability criterion for the resulting histograms. Tests on standard face recognition datasets show that this method can create compact yet discriminative descriptors.**

I. Introduction

Local visual descriptors [1] have become part of state-of-the-art systems in many areas of computer vision. Face recognition is no exception, and methods based on local descriptors have been shown to be more robust to occlusion, misalignment and moderate pose changes than traditional holistic methods such as [2], [3]. Comparisons of methods based on local descriptors for face recognition may be found in [4] and [5].

A particularly simple and successful local visual descriptor for face recognition is the histogram of LBPs as introduced by [6]. Since their introduction, several variations have appeared, such as local ternary patterns [7], elongated local binary patterns [8], multi-scale LBPs [9], patch-based LBPs [10], center symmetric LBPs [11], LBPs on Gabor-filtered images [12], [13], and LBPs on histograms of gradients [14], to cite a few.

The main contribution of this paper is a simple method to automatically learn a discriminative, compact LBP-like descriptor from the data. The method represents LBP-like descriptors as a set of pixel comparisons within a small region, and seeks from among all the possible comparisons a subset that maximizes the discriminativity of the output histograms as measured by a Fisher class separability criterion. This method avoids hand-crafted designs (e.g. [9], [11]) and is supervised, unlike e.g. [15], [16]. It thus follows in the steps of [17], but is an improvement in various ways, as we describe in more detail below. Since the method explicitly searches for discriminative patterns, we dub it Discriminative Local Binary Patterns (DLBP). As a testing scenario, we consider the traditional task of *closed set face identification*. Under this task, we are given a gallery of identified face images, such that, for any unidentified probe image, the goal is to return one of the identities from the gallery.

This paper is organized as follows. Section I-A outlines the face recognition pipeline. Section II describes in detail the DLBP descriptor and contrasts it to the traditional LBP descriptor. Section III discusses related approaches. Section IV describes the results on two standard benchmark datasets. Finally, section V presents the main conclusions of this work.

A. Face recognition pipeline

Our face recognition pipeline is similar to the one proposed in [6], but we incorporate a more sophisticated illumination normalization step [7]. Figure (1) summarizes its operation, given by the following main steps:

1) Crop the face region and align the face by mapping the eyes to a canonical location with a similarity transform.
2) Normalize illumination with Tan and Triggs' [7] Difference of Gaussians (DoG) filter.
3) Partition the face image in a grid with equally sized cells, the size of which is a parameter.
4) For each grid cell, apply a feature extraction operator (such as LBPs or DLPBs, as described below) to each pixel in the grid cell. Afterward, create a histogram of the feature values and concatenate these histograms into a single vector, usually known as "spatial histogram".
5) Classify a probe face with the identity of the nearest neighbor in the gallery, where the nearest neighbor distance is calculated with the (possibly weighted) L_1 distance between the histograms of the corresponding face images. In our algorithm, we use one weight for each grid cell. That is, the distance between two spatial histograms $S^1 = (H^1_1, \cdots, H^1_M)$ and $S^2 = (H^2_1, \cdots H^2_M)$ is

$$dist(S^1, S^2) = \sum_{m=1}^{M} w_m \sum_i |H^1_{mi} - H^2_{mi}| \quad (1)$$

where H_{mi} is the ith bin of the mth histogram. We specify how the weights are obtained below.

II. Local Binary Pattern descriptors

A. Traditional Local Binary Patterns

Local binary patterns were introduced by Ojala et al [18] as a fine scale texture descriptor. In its simplest form, an LBP description of a pixel is created by thresholding the values of a 3×3 neighborhood with respect to its central pixel and interpreting the result as a binary number.

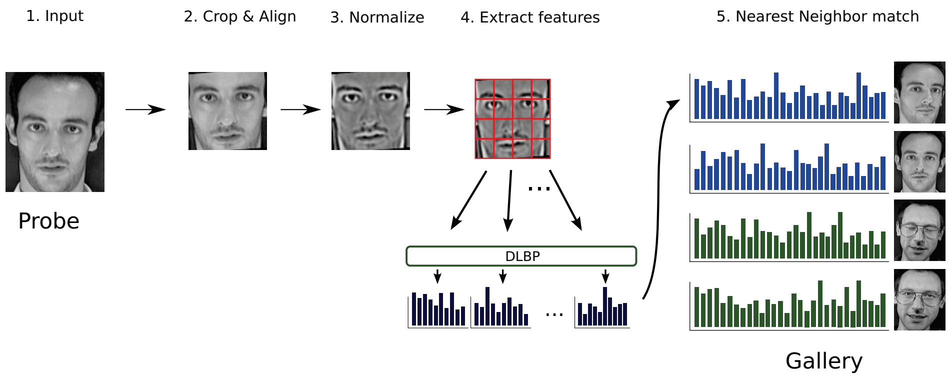

Fig. 1. The face recognition pipeline.

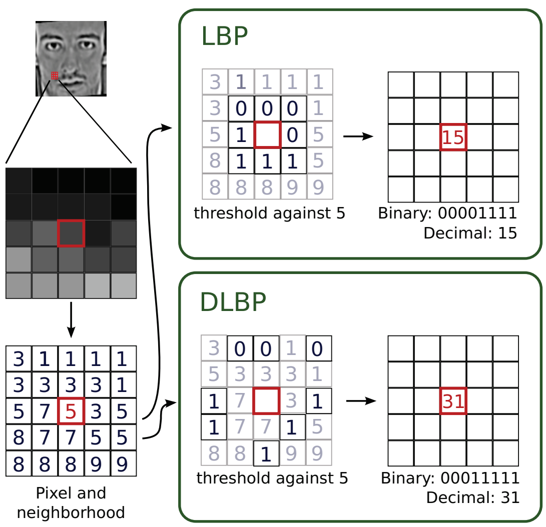

Fig. 2. The LBP operator versus the DLBP operator. In the LBP operator, pixel comparisons are restricted to a predetermined pattern; in DLBP, the pattern is learned discriminatively.

In a more general setting, an LBP operator assigns a decimal number to a pair $(c, \mathbf{n})$,

$$b = \sum_{i=1}^{S} 2^{i-1} I(c, n_i)$$

where c represents a center pixels, $\mathbf{n} = (n_1, \dots n_S)$ corresponds to a set of pixels sampled from the neighborhood of c according to a given pattern, and

$$I(c, n_i) = \begin{cases} 1 & \text{if } c < n_i \\ 0 & \text{otherwise} \end{cases}$$

This can be seen as assigning a 0 to each neighbor pixel in $\mathbf{n}$ that is smaller than the center pixel c, a 1 to each neighbor larger than c, and interpreting the result as a number in base 2. In this way, for the case of a neighborhood of S pixels, there are 2^S possible LBP values. Figure 2 illustrates the operation of LBPs and DLBPs (explained below).

B. Our approach: Discriminative local binary patterns

Traditionally, the neighbor pixels are sampled in a circular shape that is parameterized by S and r, the radius in pixels of the circle. Thus, the 3×3 descriptor corresponds to $S = 8$ and $r = 1$. But other configurations are possible. For example, [8] proposes the use of elliptical shapes, parameterized by the length of the axes, and [10] propose a pattern with two rings. In place of these hand-crafted shapes we propose to learn the best patterns in a supervised fashion. In particular, the set of neighbors $\mathbf{n} = (n_1, \dots, n_S)$ is not determined by a parametric form but may correspond to arbitrary pixels within a small distance from the center, as seen in figure 2. Thus, the space of possible patterns in our method is determined by two parameters: r and S. r is the size of the neighborhood, and has a different meaning than in LBP; the neighbors n_i must be within a square neighborhood centered on c, but not necessarily at distance r (see figure 3). S, as in LBP, is the number of samples. In general, a larger S results in better classification accuracy, but has a larger cost in computation and storage.

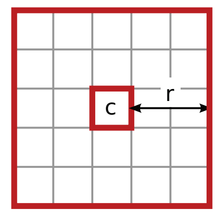

Fig. 3. Pixel neighborhood used in DLBP. The inner square is the center pixel c, and the neighborhood corresponds to all the pixels enclosed in the larger square. The size of the neighborhood is determined by the radius r.

Within the square neighborhood given by r, there are $(2r + 1)^2 - 1$ possible pixel comparisons. We wish our DLBP operator to consist of a subset $\mathbf{n}$ of those comparisons of size S that maximizes the discriminativity of the output histograms. To quantify discriminativity we use a Fisher-like class separability criterion:

$$J = \frac{(\mu_w - \mu_b)^2}{\sigma_w^2 + \sigma_b^2} \tag{2}$$

where μ_w and μ_b are the mean within-subject and between-subject distances of the histograms induced by the set $\mathbf{n}$, and σ_w^2 and σ_b^2 are the variances of the within-subject and between-subject distances of the histograms induced by the set. This criterion was used by Zhang et al [12] to weigh different facial regions.

Unfortunately, finding the set $\mathbf{n}$ of comparisons that maximizes J subject to the size constraint of S is an intractable combinatorial optimization problem. Therefore we use a simple iterative heuristic algorithm, stochastic hill climbing, to obtain an approximate solution. Other heuristics, such as simulated annealing or Tabu search could also be used. Informally, the algorithm begins with a random solution (a random set of pixel comparisons, in our case) and iteratively attempts to improve it by making small modifications (swapping a pixel comparison by a different one). In pseudo code, the hill climbing procedure is

```
proc hillclimb(set) ≡
  J* := −inf
  for i := 1 to hill_climbing_iterations do
      new_set := copy(set)
      J' := J(new_set)
      for j := 1 to tweaks do
          set' := tweak(copy(set))
          if J(set') ≥ J'
            then
                  J' := J(set')
                  new_set := set'
          fi
      od
      set := new_set
      if J' ≥ J*
        then
              best := set
              J* := J'
      fi
  od
  return best
.
proc J(set) ≡
  Quantize data into histograms with set
  evaluate and return (μ_w − μ_b)^2 / (σ_w^2 + σ_b^2)
.
proc tweak(set) ≡
  Select random pixel comparison n_i from set
  Set n_i to another random pixel within the neighborhood
   that does not already belong to set
  return set
.
```

We run this algorithm five times and store the best set obtained during the hill climbing phase along with its associated J^* value.

Since we expect different patterns to be discriminative in different face regions, we learn a new DLBP for each grid cell. One of the side benefits of the optimization scheme is that we may use the J^* obtained for each cell as a weight for the distance calculation in (1), assuming that J^* reflects how discriminative is the facial region corresponding to the grid cell.

The optimization process has its own set of parameters, namely the number of hill climbing iterations and the number of tweaks tested at each hill climbing iteration. These are dictated mostly by the available computing resources. We use 60 hill climbing iterations and 30 tweaks tested per iteration. With these parameters, our C++ implementation of the training process takes around 4 hours (in total, for all face regions) on a 1.6GHz laptop with $S = 8$.

III. Related work

Our method can be seen as a simplification and an improvement to Decision Tree-Local Binary Patterns (DTLBP) [17]. This method models LBPs as trees and uses an ID3-like [19] algorithm to learn a discriminative LBP-like descriptor. DLBPs use a simpler set of pixel comparisons in place of the tree, which is faster and simpler to learn. In addition, optimizing the Fisher separability criterion results in better classification accuracy with the nearest neighbor method than the entropy gain criterion used by ID3.

The simplification of trees to sets of comparisons is analogous to the simplification of random trees to random Ferns proposed for the task of keypoint matching in [20]. The authors observe that Ferns give similar results to trees but have a smaller computational complexity. Though Ferns are similar to our DLBPs, DLBPs use single pixel comparisons instead of planes and do not use the seminaive Bayesian method proposed in [20].

Ahonen et al [21] proposed to view the difference $c - n_i$ of each neighbor pixel n_i with the center as the response of a particular filter centered on c. Under this view, the LBP operator is a coarse way to quantize the joint responses of various filters (one for each neighbor n_i). Likewise, DLBP can also be seen as a quantizer of these joint responses, but it is built adaptively and discriminatively.

Trees have become a popular quantization method in computer vision. Moosmann et al [22] use Extremely Randomized Clustering forests to create codebooks of SIFT descriptors [23]). Shotton et al. [24] use random forests to create codebooks for use as features in image segmentation. While the use of trees in these works is similar to ours, they use the results of the quantization in a very different way; the features are given to classifiers such as SVMs, which are not suitable for use in our problem.

Wright et al. [25] use unsupervised random forests to quantize SIFT-like descriptors for face recognition. The main difference with our algorithm is that we do not quantize complex descriptors extracted from the image. In addition, the accuracy of their algorithm on the tested datasets is relatively poor compared to other state-of-the-art algorithms. This may be due to the use of an unsupervised algorithm to construct the trees.

There are various recent works using K-Means to construct codebooks to be used for face recognition in a framework similar to ours. Meng et al [26] use it to directly quantize patches from the grayscale image patches. Xie et al [16] as well as [15] use it to quantize patches from images convolved with Gabor wavelets at various scales and orientations. These algorithms are the closest in spirit to our work, since they are partly inspired by LBPs. These algorithms differ from ours in the algorithm used to construct the codebook. They use K-Means, which has the drawback of not being supervised and thus unable to take advantage of labeled data. In addition, for the same number of codes, K-Means are less efficient than DLBPs. Finally, unlike ours, two of the above algorithms incorporate Gabor wavelet features; the cost of convolving the image with the real and imaginary parts of 40 or so Gabor filters may be excessive for some applications.

Our method also differs from methods that use methods such as Boosting to select histograms corresponding to particular LBP windows or scales (e.g. [9], [27]) or histogram

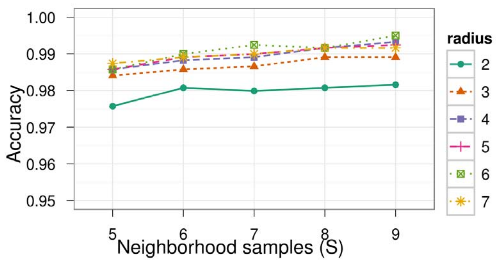

Fig. 4. Effect of neighborhood samples (S) and radius on accuracy of DLBP on FERET fb.

bins (e.g. [28], [29]). The reason is that we do not select from among LBP features that have already been extracted, but instead search for the best feature to extract. Selecting from pre-extracted features is not feasible with a large number of pixel comparisons (S), since the length of the histograms grows exponentially with S.

Another line of investigation worth mentioning is the use of heuristic algorithms, and in particular evolutionary algorithms, to construct visual descriptors for different purposes. Perez and Olague [30] use Genetic Programming (GP) with a large set of terminals to construct invariant region descriptors for visual matching. Yu and Bhanu [31] also use GP with a large set of operators and Gabor filtering to induce features for facial expression recognition. Kowaliw et al [32] use a variant of GP known as cellular GP to build features for an image classification task. Compared to these approaches, our features are simpler, since they do not use a complex set of operations and terminals.

IV. EXPERIMENTS

We perform experiments on the FERET [33] and the CAS-PEAL-R1 [34] benchmark databases. We report results with and without weights, where the weights for each region are set as the final J value from (2) for the set of each region.

Regarding the parameters, in order to give the algorithm flexibility in the choice of patterns we use a relatively large radius, $r = 7$ [1]. This was the radius used in [17]. Figure IV illustrates the effect of radius and neighborhood samples on the accuracy of DLBP on FERET fb. The trend, also seen in other datasets, is that all radii larger than 2 perform comparably. S is varied to evaluate the size-accuracy tradeoff. In all images we partition the image into an 7×7 grid, as originally used in [6]. While in general we have found this partition to provide good results, it is likely that adjusting the grid size to each database may yield better results.

For each experiment we show our results along with the best results from similar works in the recent literature: the original LBP algorithm from Ahonen [6]; the Local Gabor Binary Pattern (LGBP) algorithm [12], which applies LBP to Gabor-filtered images; the Local Visual Primitive (LVP) algorithm [26], which uses K-Means to quantize grayscale patches; the Decision Tree Local Binary Pattern (DTLBP) algorithm [17], which uses decision trees to find discriminative LBPs on grayscale images; Local Gabor Textons (LGT) algorithm [15] and the Learned Local Gabor Pattern (LLGP) [16] algorithm, which use K-Means to quantize Gabor filtered-images; and the Histogram of Gabor Phase Patterns (HGPP) algorithm [35], which quantizes Gabor filtered images into histograms that encode not only the magnitude, but also the phase information from the image. For each algorithm, if a weighting scheme is used, we show the best results with the weighting scheme under the name of the algorithm followed by '-W'. We also show the results of using a purely random set of pixel comparisons as RLBP (for Random LBP) to assess the effects of the supervised optimization. For the LBP algorithm, we give the accuracy obtained by the original authors as well as by our own implementation. The reason is that due to the different preprocessing and border handling the accuracies differ. In addition, for LBP we add results with a radius of 7, since DLBP uses a radius of that scale, and also add results with the weight given by the same Fisher J value as in DLBP. We also show results with and without Tan-Triggs normalization to show the effect this step has on the results.

The results cited from other papers are not strictly comparable, since there may be differences in preprocessing, and for FERET, the training set used, but they provide a meaningful reference.

A. Results on FERET

For FERET, we use *fa* as gallery and *fb*, *fc*, *dup1* and *dup2* as probe sets. For training, we use the FERET standard training set of 762 images from the training CD provided by the CSU Face Identification Evaluation System package [36]. The results are summarized in tables I and II.

We can see that our algorithm does well on FERET, specially when normalization is used. Without normalization, DLBP's accuracy suffers on fc, which varies illumination. With Tan-Triggs normalization it obtains the best results on fb, $dup1$ and $dup2$, and comparable to the best on fc.

B. Results on CAS-PEAL-R1

In CAS-PEAL-R1 we use the standard training and gallery subsets, and we use the Expression, Lighting and Accessory subsets as probes. The results are summarized in tables III and IV.

In this dataset our algorithm also does well. It obtains the best results in the Expression and Accessory datasets, tying with HGPP in the latter. As before, without normalization the performance of DLBP suffers on datasets with illumination variation. On the lighting dataset, the overall performance of all the algorithms is rather poor. In this case, the best performance are given by LGBP, HGPP and LLGP. All these algorithms use features based on Gabor wavelets, which suggests that Gabor features provide robustness against the extreme lighting variations in this dataset.

[1] To handle the border pixels, we simply added a black border of width 7 to each image. More sophisticated schemes gave similar or worse results.

Method	No TT fb	fc	dup1	dup2	With TT fb	fc	dup1	dup2
LBP_8^2	.96	.53	.60	.40	.93	.96	.72	.67
LBP_8^7	.96	.34	.61	.45	.98	.96	.79	.78
RLBP_5^7	.94	.28	.57	.35	.95	.93	.71	.66
RLBP_6^7	.95	.28	.57	.36	.96	.95	.75	.75
RLBP_7^7	.96	.38	.59	.37	.97	.95	.78	.73
RLBP_8^7	.96	.32	.60	.39	.97	.97	.80	.75
RLBP_9^7	.97	.44	.61	.42	.98	.97	.83	.81
DLBP_5^7	.96	.29	.61	.40	.97	.94	.77	.75
DLBP_6^7	.97	.31	.61	.41	.98	.97	.80	.79
DLBP_7^7	.98	.39	.63	.44	.98	.98	.81	.80
DLBP_8^7	.98	.37	.64	.47	.98	**.99**	.84	.82
DLBP_9^7	.98	.40	.66	.48	**.99**	**.99**	.85	.84
$\mathrm{LBP\text{-}W}_8^2$	.98	**.54**	.62	.45	.97	.97	.72	.68
$\mathrm{LBP\text{-}W}_8^7$	.98	.32	.65	.53	**.99**	.97	.81	.80
$\mathrm{RLBP\text{-}W}_5^7$	.97	.27	.59	.41	.97	.93	.72	.69
$\mathrm{RLBP\text{-}W}_6^7$	.98	.30	.61	.45	.98	.94	.77	.76
$\mathrm{RLBP\text{-}W}_7^7$	.98	.36	.63	.48	**.99**	.97	.79	.74
$\mathrm{RLBP\text{-}W}_8^7$	.98	.31	.65	.53	**.99**	.97	.81	.78
$\mathrm{RLBP\text{-}W}_9^7$	.98	.38	.65	.54	**.99**	.98	.81	.79
$\mathrm{DLBP\text{-}W}_5^7$	.98	.28	.63	.47	.98	.94	.78	.78
$\mathrm{DLBP\text{-}W}_6^7$	**.99**	.34	.63	.49	**.99**	.98	.82	.81
$\mathrm{DLBP\text{-}W}_7^7$	**.99**	.42	.66	.53	**.99**	.98	.85	**.85**
$\mathrm{DLBP\text{-}W}_8^7$	**.99**	.41	.67	.54	**.99**	**.99**	.85	**.85**
$\mathrm{DLBP\text{-}W}_9^7$	**.99**	.48	**.68**	**.55**	**.99**	**.99**	**.86**	**.85**

TABLE I

ACCURACY ON FERET PROBE SETS. DLBP_d^r CORRESPONDS TO A SET OF SIZE d AND RADIUS r. "-W" INDICATES WEIGHTS. "NO TT" INDICATES NO ILLUMINATION NORMALIZATION.

Method	fb	fc	dup1	dup2
LBP [6]	.93	.51	.61	.50
LGBP [16]	.94	.97	.68	.53
LVP [16]	.97	.70	.66	.50
LGT [15]	.97	.90	.71	.67
HGPP [35]	.98	.99	.78	.76
LLGP [16]	.97	.97	.75	.71
LBP-W [6]	.97	.79	.66	.64
LGBP-W [16]	.98	.97	.74	.71
LVP-W [16]	**.99**	.80	.70	.60
HGPP-W [35]	.98	.99	.78	.77
LLGP-W [16]	**.99**	.99	.80	.78
DT-LBP [17]	**.99**	**1.0**	.84	.80
$\mathrm{DLBP\text{-}W}_9^7$ (ours)	**.99**	.99	**.86**	**.85**

TABLE II

COMPARISON OF ACCURACY WITH OTHER ALGORITHMS ON FERET PROBE SETS. FIGURES FROM ENTRIES WITH CITATIONS COME FROM THE RESPECTIVE CITATION.

C. *Discussion*

The results show that DLBPs are highly discriminative features. However, it seems that without normalization the learning process tends to overfit on datasets with intense illumination variation and performance suffers. It should be noted that the normalization is a computationally inexpensive process; it consists in convolution with a Difference of Gaussians filter and a couple of equalization steps. This is much faster than convolution with the real and imaginary parts of 40 Gabor filters, as in LGBP, LGT and HGPP.

As expected, the supervised optimization process improves upon the purely random descriptor, though in some cases the difference is relatively small. Thus RLBP could be of interest for unsupervised scenarios.

We highlight the the ability our algorithm to create compact yet discriminative descriptors. Even when using $S = 5$, which yields histograms of size 32, it performs comparably or better than (non-uniform) LBP, of size 256. Our largest and best-performing descriptor ($S = 9$) yields histograms of size 512, which are smaller than those used by Gabor-based approaches that concatenate histograms for each Gabor orientation and scale; for example, LLGP uses 12 Gabor filters and a codebook of $K = 70$ for each, giving histograms of size $12 \times 7 = 840$; LGBP [12] uses 40 Gabor images and an LBP of $S = 8$ for each, resulting in histograms of size $256 \times 40 = 10240$. DLBP's histograms are also much smaller than the ones used in [17]; the results we show are from a descriptor with histograms of size 2^{13}.

V. CONCLUSIONS AND FUTURE WORK

We have proposed a novel method that uses training data to learn compact and discriminative LBP-like descriptors. The algorithm obtains encouraging results on standard databases, and presents better results that several state-of-the-art alternative solutions. In particular, with respect to a face recognizer based on the widely used LBPs, our approach presents an increase in accuracy, demonstrating the advantages of using an adaptive and discriminative set of local binary patterns.

Regarding future work, we are evaluating how to increase the robustness of the descriptor to datasets with intense illumination variation.

VI. ACKNOWLEDGMENTS

This work was partially funded by FONDECYT grant 1095140 and LACCIR Virtual Institute grant No. R1208LAC005 (`http://www.laccir.org`).

Portions of the research in this paper use the FERET database of facial images collected under the FERET program.

The research in this paper uses the CAS-PEAL-R1 face database collected under the sponsorship of the Chinese National Hi-Tech Program and ISVISION Tech. Co. Ltd.

REFERENCES

[1] K. Mikolajczyk and C. Schmid, "Performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–30, 2005.

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

[3] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linearprojection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, 1997.

[4] J. Ruiz-del-Solar, R. Verschae, and M. Correa, "Recognition of faces in unconstrained environments: A comparative study," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, pp. 1–20, 2009.

[5] J. Zou, Q. Ji, and G. Nagy, "A comparative study of local matching approach for face recognition," *IEEE Trans. Image Process.*, vol. 16, no. 10, pp. 2617–2628, 2007.

Method	No TT Exp.	No TT Acc.	No TT Light.	With TT Exp.	With TT Acc.	With TT Light.
LBP_8^2	.95	**.82**	.19	.97	.89	.29
LBP_8^7	.94	.72	.17	.94	.85	.27
$RLBP_5^7$	.90	.65	.14	.91	.81	.24
$RLBP_6^7$	.92	.69	.15	.91	.82	.23
$RLBP_7^7$	.93	.7	.15	.93	.84	.25
$RLBP_8^7$	.94	.75	.16	.95	.86	.27
$RLBP_9^7$	.93	.72	.17	.95	.87	.28
$DLBP_5^7$	.94	.73	.18	.96	.88	.31
$DLBP_6^7$	.95	.77	.20	.97	.90	.35
$DLBP_7^7$	.96	.79	.21	.97	.90	.36
$DLBP_8^7$	**.97**	.81	.22	.98	.91	.39
$DLBP_9^7$	**.97**	**.82**	.23	.98	**.92**	.40
$LBP\text{-}W_8^2$	**.97**	.78	.20	.97	.89	.31
$LBP\text{-}W_8^7$	.94	.71	.18	.93	.86	.26
$RLBP\text{-}W_5^7$	.89	.65	.15	.91	.81	.24
$RLBP\text{-}W_6^7$	.92	.67	.17	.92	.83	.23
$RLBP\text{-}W_7^7$	.94	.69	.16	.93	.85	.24
$RLBP\text{-}W_8^7$	.94	.75	.18	.95	.87	.27
$RLBP\text{-}W_9^7$	.93	.74	.18	.94	.88	.28
$DLBP\text{-}W_5^7$	.95	.72	.20	.96	.87	.31
$DLBP\text{-}W_6^7$	.95	.76	.22	.97	.90	.35
$DLBP\text{-}W_7^7$	.96	.78	.22	.98	.90	.36
$DLBP\text{-}W_8^7$	**.97**	.81	.23	.98	**.92**	.40
$DLBP\text{-}W_9^7$	**.97**	**.82**	**.24**	**.99**	**.92**	**.41**

TABLE III

ACCURACY ON CAS-PEAL-R1 PROBE SETS. $DLBP_d^r$ CORRESPONDS TO A SET OF SIZE d AND RADIUS r. "-W" INDICATES WEIGHTS. "NO TT" INDICATES NO ILLUMINATION NORMALIZATION.

Method	Expression	Accessory	Lighting
LGBP [16]	.95	.87	.51
LVP [26]	.96	.86	.29
HGPP [35]	.96	**.92**	.62
LLGP [16]	.96	.90	.52
LVP-W [26]	.96	.86	.33
HGPP-W [35]	.97	**.92**	**.63**
LLGP-W [16]	.96	**.92**	.55
DT-LBP [17]	.98	**.92**	.41
$DLBP\text{-}W_9^7$ (ours)	**.99**	**.92**	.41

TABLE IV

COMPARISON OF ACCURACY WITH OTHER ALGORITHMS ON CAS-PEAL-R1 PROBE SETS. FIGURES FROM ENTRIES WITH CITATIONS COME FROM THE RESPECTIVE CITATION.

[6] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, 2006.

[7] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–50, Jun. 2010.

[8] S. Liao and A. C. S. Chung, "Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude," in *ACCV*, Berlin, Heidelberg, 2007, pp. 672–679.

[9] S. Liao, X. Zhu, Z. Lei, L. Zhang, and S. Li, "Learning multi-scale block local binary patterns for face recognition," in *Advances in Biometrics*, 2007, pp. 828–837.

[10] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Real-Life Images Workshop at ECCV*, October 2008.

[11] M. Heikkil, M. Pietikinen, and C. Schmid, "Description of interest regions with local binary patterns," *Pattern Recognition*, vol. 42, no. 3, pp. 425–436, 2009.

[12] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang, "Local gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition," in *ICCV*, 2005.

[13] S. Xie, S. Shan, X. Chen, and W. Gao, "V-LGBP: Volume based Local Gabor Binary Patterns for face representation and recognition," in *ICPR*, 2008.

[14] N.-S. Vu and A. Caplier, "Face recognition with patterns of oriented edge magnitudes," in *ECCV*, vol. 6311, 2010, pp. 313–326.

[15] Z. Lei, S. Li, R. Chu, and X. Zhu, "Face recognition with local Gabor textons," *Advances in Biometrics*, pp. 49–57, 2007.

[16] S. Xie, S. Shan, X. Chen, X. Meng, and W. Gao, "Learned local gabor patterns for face representation and recognition," *Signal Processing*, vol. 89, no. 12, pp. 2333 – 2344, 2009.

[17] D. Maturana, D. Mery, and A. Soto, "Face recognition with decision tree-based local binary patterns," in *ACCV*, 2010.

[18] T. Ojala, M. Pietikinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[19] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.

[20] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua, "Fast keypoint recognition using random ferns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 448–461, 2010.

[21] T. Ahonen and M. Pietikinen, "Image description using joint distribution of filter bank responses," *Pattern Recognition Letters*, vol. 30, no. 4, pp. 368 – 376, 2009.

[22] F. Moosmann, E. Nowak, and F. Jurie, "Randomized clustering forests for image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 1632–1646, 2008.

[23] D. G. Lowe, "Distinctive image features from Scale-Invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[24] J. Shotton, M. Johnson, and R. Cipolla, "Semantic texton forests for image categorization and segmentation," in *CVPR*, 2008.

[25] J. Wright and G. Hua, "Implicit elastic matching with random projections for pose-variant face recognition," in *CVPR*, 2009, pp. 1502–1509.

[26] X. Meng, S. Shan, X. Chen, and W. Gao, "Local visual primitives (LVP) for face modelling and recognition," in *ICPR*, 2006.

[27] G. Zhang, X. Huang, S. Li, Y. Wang, and X. Wu, "Boosting local binary pattern (lbp)-based face recognition," in *Advances in Biometric Person Authentication*, ser. Lecture Notes in Computer Science, vol. 3338. Springer Berlin / Heidelberg, 2005, pp. 179–186.

[28] C. Shan and T. Gritti, "Learning discriminative LBP-histogram bins for facial expression recognition," in *BMVC*, 2008.

[29] X. Wang, C. Zhang, and Z. Zhang, "Boosted multi-task learning for face verification with applications to web image and video search," in *CVPR*, 2009.

[30] C. B. Perez and G. Olague, "Learning invariant region descriptor operators with genetic programming and the f-measure," in *ICPR*, 2008, pp. 1–4.

[31] J. Yu and B. Bhanu, "Evolutionary feature synthesis for facial expression recognition," *Pattern Recogn. Lett.*, vol. 27, no. 11, pp. 1289–1298, 2006.

[32] T. Kowaliw, W. Banzhaf, N. Kharma, and S. Harding, "Evolving novel image features using genetic programming-based image transforms," in *CEC*, 2009, pp. 2502–2507.

[33] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for Face-Recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, 2000.

[34] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao, "The CAS-PEAL large-scale Chinese face database and baseline evaluations," *IEEE Trans. Syst., Man, Cybern. A*, vol. 38, no. 1, pp. 149–161, 2008.

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

[36] D. Bolme, J. Beveridge, M. Teixeira, and B. Draper, "The CSU face identification evaluation system: Its purpose, features and structure," in *ICCV*, 2003.