

Weighty LBP: A New Selection Strategy of LBP Codes Depending on Their Information Content

Maria De Marsico¹ and Daniel Riccio²(✉)

¹ Sapienza University of Rome, Rome, Italy
demarsico@di.uniroma1.it

² University of Naples Federico II, Naples, Italy
daniel.riccio@unina.it

Abstract. This paper presents a novel variation of the use of LBP codes. Similarly to Uniform LBP and Local Salient Patterns (LSP), it aims at both obtaining an effective texture description, and decreasing the length of the feature vectors, i.e., of the chains of LBP histograms. Instead of considering uniform codes, we rather consider the codes providing the highest “representativeness” power with respect to texture features. We identify this subset of codes by a generalized notion of entropy. This allows determining the most informative items in an homogeneous set.

Keywords: Local binary patterns · Information content · Entropy

1 Introduction

Though quite simple and light to compute, Local Binary Patterns (LBP) represents a very efficient texture operator. The basic procedure uses the value of each image pixel in turn as a binarization threshold for the values in its neighborhood (originally a 3×3 window); afterward, the code assigned to each pixel is the binary number represented by the string of binary elements obtained in such neighborhood. While the gray-level value of a given pixel represents its spectral propriety, its LBP code represents the textural aspect of the given pixel. LBP has achieved a great and still increasing popularity since its introduction in [9], where it is presented as a simplification of texture units (TUs) [18] making up the texture spectrum of an image. Similarly to LBP, TUs are obtained from a neighborhood of 3×3 pixels, yet using three (0, 1, 2) instead of two values, giving a much higher number of codes. The texture spectrum is defined as the histogram (frequency of occurrences) of texture units computed over a region. The work in [9] shows that LBP, when used together with a simple local contrast measure, achieves better performance in unsupervised texture segmentation than other methods for texture analysis quite popular at that time. Due to this descriptive power, since its introduction, LBP has been the object of extensive investigations and evaluations, as well as variations [10]. It has been applied to address many problems, in particular, in the field of biometrics. A few examples include face recognition [1], demographics classification [19], gender recognition [15, 17],

or even face expression recognition [14]. A comprehensive survey of the use of LBP in Computer Vision can be found in [11].

This work deals with a novel approach to reduce the size of the code set for LBP, along the line of Uniform LBP codes [16] and Local Salient Patterns (LSP) [2]. Therefore, these two are the reference techniques we will compare with. In other words, the contribution of this work is to propose a new code reduction technique, and to compare it with the previously proposed ones.

The new operator is denoted as Weighty LBP (W-LBP). The core idea is to choose the most relevant codes according to their informative content with respect to textural features. Though Uniform LBP codes can be perceptually appreciated as more relevant with respect to the remaining ones, there is no assessment of their individual “informative” power. The same holds for LSP. We propose to analyze the LBP codes by exploiting a generalized definition of entropy. This was introduced to identify relevant face images in a set, and used for image analysis [6], for template selection in video-surveillance tasks [4], for the construction of a difference space for face image classification [13], and also for clustering [8]. Then it was extended to analyze generic items in a set, e.g., to quantize colors for image segmentation [5]. We apply the underlying approach here to select the most “representative” LBP codes.

The application of the method based on set entropy requires to exploit a suited similarity measure able to capture the characteristic nature of the items at hand (see Sect. 3.1). In the case of LBP such items are binary strings. Despite many different similarity/distance measures have been proposed, each such measure captures different aspects. Therefore, in the following we will briefly present the ones that we chose to analyze in order to capture the possible “informative power” of LBP codes. Afterward, we will follow a double evaluation procedure. In the first place, for each similarity measure, the subset of the most representative codes will be identified, and then among the obtained subsets those achieving the best classification results will be selected.

2 Related Work

LBP can be used in two ways. It can be used to characterize an image by a feature vector built with the histogram of LBP codes from the image, or to produce a gray level *feature image* by substituting each pixel in the original image by its LBP code. Feature images then undergo further computer vision processing. In particular applications, e.g., face recognition, LBP robustness can be increased by processing the image divided in cells according to a grid, whose size depends on image resolution. In this case, LBP is applied separately to each grid cell, and, in the case of histograms, the final feature vector is obtained by chaining the single cell histograms. This causes possibly huge feature vectors. Methods requiring a training step may incur the curse of dimensionality problem. For this reason, an interesting research line investigates how to identify and use a reduced the number of LBP codes, achieving a possibly better texture characterization with shorter feature vectors. However, finding the optimal subset of patterns is

a demanding combinatorial problem. The selection of a subset of NP or less patterns out of the total 256, requires to assess the performance of a number of possible solutions that even for moderate values of NP requires huge computing resources. Therefore a suboptimal yet satisfying solution is often searched for. The work in [16] compares two approaches to extract a relevant subset of LBP codes. The first one uses beam search and explores subsets of patterns minimizing the classification error. The method iteratively increases the size of the pattern subset up to dimension NP and updates a list of the best BS subsets identified. The classification at each iteration exploits a reduced LBP histogram that contains one bin for each pattern chosen so far. All the remaining patterns are collapsed into a single bin. After NP iterations, the procedure returns BS distinct pattern sets, from which the optimal patterns can be chosen.

The second approach proposes the nowadays popular Uniform LBP patterns. It first defines a measure of *nonuniformity* $U(\text{LBP})$, which corresponds to the number of transitions (from 0 to 1 or the inverse) in the circular bitwise representation of the code. The assumption is that the lower the number of transitions, the more robust the code to image distortions. Based on this, the authors propose using the nine uniform patterns and their circularly rotated versions (this allows some transformation invariance). In practice, this corresponds to use 58 out of the 256 original unrotated patterns. Even in this case, all the remaining patterns are compressed into a single bin, therefore obtaining a 59-bin histogram. The conclusions drawn in [16] underline that every application may have its optimal set of patterns, but uniform patterns appear to perform well in many situations.

An example of a different strategy to address the reduction of LBP codes is represented by Local Salient Patterns (LSP) [2]. This recent approach derived from the original formulation of LBP focuses on the location of the largest positive and negative differences within the pixel neighborhood. This is deemed to remove the noise influence. The coding takes into account the possible pairs of neighbor indexes ($p_{diffmax}, p_{diffmin}$) that provide the maximum and the minimum difference with the central value of the neighborhood (usually, a 3×3 window). Therefore there are 57 distinguished codes (the last one corresponds to equal differences for all neighbors). This descriptor has achieved good performances in different facial analysis tasks, and experiments reported in [2] show that in most cases, LSP can outperform Uniform LBP.

3 The Proposed LBP Reduction

3.1 Entropy to Select Representatives in a Set

In image analysis, entropy is usually exploited as a measure of randomness/homogeneity of image pixels. Each pixel x in image I is treated as a symbol in the alphabet emitted by a source S . As for gray scale images, the alphabet is the set of 8-bit integers in $[0, 255]$. After normalizing the image histogram values in the range $[0, 1]$, each bin represents the probability of occurrence of the corresponding symbol in I . Entropy $H(I)$ is:

$$H(I) = - \sum_{k=0}^{255} p(k) \log_2(k) \quad (1)$$

Equation 1 can be generalized to express the amount of homogeneity in a set of items of any kind, given a suited abstraction. We summarize here the basic notation. More details can be found in [6].

Given a set G of objects/elements/observations (items from here on), we first search for a suitable similarity measure s , which is used to associate a real scalar value to any pair of feature vectors (templates) used to characterize the items of interest according to a chosen set of characteristics. The choice of the similarity measure to exploit depends in the first place on the kind of items to compare, and on the extracted feature templates. Computational cost of measuring this similarity can provide a further criterion. Popular examples are Euclidean distance, if feature vectors are represented as points in a space, or Dynamic Time Warping (DTW) for time series. The noticeable property of the following definitions is that they hold whichever is the chosen similarity measure s . From here on, if not otherwise specified, the notation will identify templates with the items they describe. In a preliminary definition step, let us assume to compare the probe template v to classify with a set templates g_i (from now on denoted as *gallery*.) This produces a similarity measure $s(v, g_i)$, denoted as s_i . After score normalization, s_i is a real value in the interval $[0, 1]$. The score s_i can be interpreted as the probability that v conforms (adapts) to g_i , therefore obtaining a probability distribution over the set G , i.e., $s_{i,v} = p(v \approx g_i)$. In order to compute a total value for the entropy of the set G each of its elements is considered in turn as a probe v , to compute all-against-all similarities. Let's denote as Q the number of pairs $\langle q_i, q_j \rangle$ in G such that $s_{i,j} > 0$, used as a normalization factor; then entropy of G is denoted as $H(G)$ and computed as:

$$H(G) = -\frac{1}{\log_2(|Q|)} \sum_{q_{i,j} \in Q} s_{i,j} \log_2(s_{i,j}) \quad (2)$$

The value of $H(G)$ can be considered as a measure of heterogeneity for the items in G . It is possible to order all of them according to their *informative power* or *representativeness*, by computing their contribution to $H(G)$. Given G , the devised procedure computes the complete similarity matrix M and the value for $H(G)$. For each item $g_i \in G$, M is used to compute the value of $H(G \setminus g_i)$ obtained by ignoring g_i . The item g_i , achieving the minimum difference $f(G, g_i) = H(G) - H(G \setminus g_i)$, is selected; the matrix M is updated by deleting the i -th row and column, and the process is repeated, until all elements of G have been selected. According to this procedure, we first select the most representative samples, i.e. those causing the lower entropy decrease. The algorithm progressively reduces the inhomogeneity of the set. We finally obtain an ordering of the elements as they are selected by the algorithm, with the corresponding value of $f(\cdot)$. The trend of the resulting curve presents local maxima and minima in a smooth saw tooth shape, that can be usually quite well approximated by a parabola (see [6]). The values obtained for $f(\cdot)$ can be used to cluster the set elements, in a way similar to one of the approaches in [13]. The first relative maximum becomes the representative element of the first cluster. The following elements along $f(\cdot)$ until the next relative maximum are included in this cluster. A new cluster is created

when the next relative maximum is found, and cluster population is continued as before. This procedure is repeated till the end of $f(\cdot)$.

3.2 Binary Similarity and Distance Measures

In order to explore the information content of LBP codes with respect to different similarity measures, we refer to the survey presented in [3]. In that work, 76 binary similarity and distance measures are discussed that have been used over the last century, and their correlation is investigated through a hierarchical clustering technique. The interested reader can refer to that paper. For our purposes, we selected a subset of 65 out of the measures mentioned there, leaving out or merging duplicates. Similarly to [3], the definitions of measures are expressed by Operational Taxonomic Units (OTUs) [7]. Assume to have two binary vectors, i and j . Let n be their common dimension. The following notation is used:

- a = the number of vector entries where the values of i and j are both 1 (or *presence*, if the binary values are interpreted in this way), meaning positive matches: $a = i \bullet j$
- b = the number of entries where the value of i and j is (0, 1) (or *i absence mismatch*): $b = \bar{i} \bullet j$
- c = the number of attributes where the value of i and j is (1, 0) (or *j absence mismatch*): $c = i \bullet \bar{j}$
- d is the number of attributes where both i and j have 0 (or *absence*), meaning negative matches: $d = \bar{i} \bullet \bar{j}$.

The sum $a + d$ gives the total number of matches between i and j , while the sum $b + c$ gives the total number of mismatches between i and j . Measures defined as distances were transformed into similarities to obtain consistent measures.

4 Experimental Results

The experiments carried out for this work aimed at investigating a novel strategy to identify the most “informative” binary patterns produced by the LBP feature extractor, and how reducing the LBP code set to them can affect the performance of a classifier in terms of recognition accuracy. In the specific case, the experiments adopted a very simple classifier, namely Nearest Neighbor (NN) in order to avoid the dependence of the observed variations from factors not related to the aspect under study (the different sets of LBP codes). With the same idea in mind, the face database used as testbed is EGA [12]. This dataset is the result of the integration of subsets of a number of existing face datasets, that are quite different in nature in terms of ethnicity (E), gender (G), and age (A) of subjects. While EGA is expressly built to be quite balanced with respects to such demographic traits, it also offers a good variety in terms of image quality. It includes a total of 2345 images captured from 469 subjects, 5 images per subject. More details on source datasets and EGA organization can be found in

[12]. As for this work, it is important to underline that, since EGA collects face images extracted from datasets with different characteristics, both for the demographics of subjects, for capture setting and for capture devices, carrying out experiments on it is equivalent to carrying out experiments on the corresponding subsets of the source datasets. All experiments in this work considered all EGA subjects, with two out of the five images each: the first one in the dataset for the experiment gallery and the second one as the probe. Each image was pre-processed by Viola-Jones algorithm to detect the position of the face and of the center of the eyes. Faces were resized in order to have a constant inter-eye distance of 40 pixels, and cropped to 64×100 pixels. No pre-processing was performed regarding illumination, because LBP is in itself an operator quite robust to most illumination distortions. Firstly, for each similarity measure the subset of the most representative codes was identified, and then among the obtained subsets those achieving the best classification results were selected. This is quite different from beam search, that tries to add any missing code to a candidate subset, and from Uniform LBP, that selects LBP codes based on some code pattern (e.g., uniformity). We rather try to identify “weighty” LBP codes.

For sake of space, it is not possible to report the definition of all the 65 considered measures. Table 1 only reports those providing results worth mentioning, with the number indicating the ordering used here. Such numbering is maintained to preserve the relation with Figs. 2 and 3 below that report exper-

Table 1. The considered similarity (S) and distance (D) measures.

S	INTERSECTION	a	(10)
D	EUCLID	$\sqrt{b+c}$	(14)
D	HELLINGER or CHORD	$2 * \sqrt{1 - a/\sqrt{(a+b) * (a+c)}}$	(22)
S	MOUNTFORD	$a/(0.5 * (a * b + a * c) + b * c)$	(29)
S	JOHNSON	$a/(a+b) + a/(a+c)$	(34)
S	DENNIS	$(a * d - b * c)/\sqrt{n * (a+b) * (a+c)}$	(35)
S	SIMPSON	$a/\min(a+b, a+c)$	(36)
S	FAGER & MCGOWAN	$a/\sqrt{(a+b) * (a+c)} - \max(a+b, a+c)/2$	(38)
S	BARONI-URBANI & BUSER-I	$(\sqrt{a * d} + a)/(\sqrt{a * d} + a + b + c)$	(60)

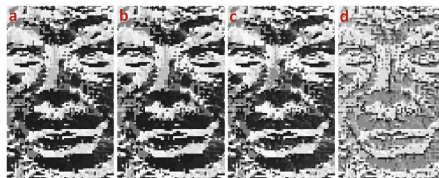


Fig. 1. LBP feature images produced with different strategies to reduce the set of codes: (a) All LBP (256 bins), (b) Uniform LBP (59 bins), (c) Entropy with MOUNTFORD measure (80 bins) and with (d) BARONI-URBANI & BUSER-I measure (41 bins).

imental results. The complete list can be found in [3] and a compacted one at the end of the paper (Table 3). Figure 1 shows some examples of LBP images produced for the same face image, but with sets of LBP codes obtained by computing representativeness according to two different measures, namely MOUNT-FORD measure (80 bins) and BARONI-URBANI & BUSER-I measure (41 bins), respectively entries indexed as (29) and (60) in Table 1. The discussion about experimental results will show that these two measures provide complementary yet orthogonal improvements. The first experiment aimed at verifying if and how a different identification of relevant LBP codes can affect the accuracy of a simple NN classifier. Classifier performances were measured in terms of Equal Error Rate (EER) in verification mode (1:1 matching with identity claim) and Recognition Rate (RR) in identification mode (1:N matching without identity claim). Chosen a similarity measure, the resulting $f(\cdot)$ function was computed and used for the clustering procedure as described in Sect. 3.1. When coding images, each code is substituted by the representative of the cluster to which it belongs. A further information provided by the used clustering algorithm is the number of returned clusters for the corresponding similarity measure. This helps evaluating also the efficiency of the produced coding (the lower the number of clusters, the lower the number of codes required), together with the obtained accuracy. Therefore, in the following figures, we have on the y axis three different items of information: the number of clusters produced, the EER and the RR value, for each of the 65 measures whose index is on x axis.

LBP feature vector is the chaining of histograms from a grid of image sub-regions, therefore a further element of interest is the size of such sub-regions (the smaller the size, the higher the number of histograms to chain, the higher the size of the final feature vector). Therefore the first experiment was repeated with four different sub-region dimensions: 16×16 , 24×24 , 32×32 , and 36×36 . Figures 2 and 3 show the plots obtained for the two extreme cases, where of course the plot of the number of clusters is always the same. The plots obtained by intermediate region sizes show a consistent trend.

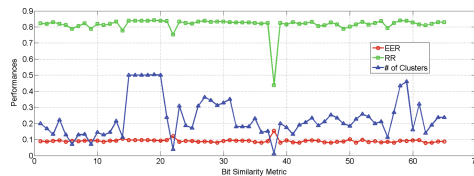


Fig. 2. Performance of NN with LBP over 16×16 sub-regions.

The results obtained in this experiment suggest that all similarity measures are affected in a generally similar way by sub-region sizes. It is possible to observe that with more information (smaller region size) both RR and EER are constantly better, and as the sub-region dimension increases, the accuracy decreases for almost all measures. RR is especially negatively affected by growing size, since

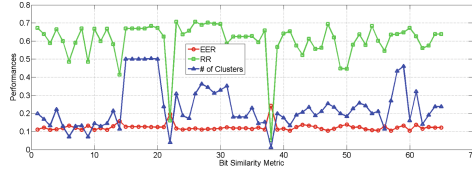


Fig. 3. Performance of NN with LBP over 36×36 sub-regions.

not only it decreases in general, but also becomes much more dependent from the exploited similarity measure. In this sense, some measures show a very different behavior from the others, either in positive or negative sense. Measures EUCLID, HELLINGER or CHORD, and FAGER & MCGOWAN, respectively (14), (22) and (38) in Table 1, generate a number of bins that is too low, taking to an excessive performance decrease, which is further accentuated whit larger regions. On the contrary, BARONI-URBANI & BUSER-I, i.e., (60) in Table 1, though producing a very low number of bins, is able to provide an accuracy comparable with the others, and is also robust to region growing. A similar result holds, though with slightly lower performance, for INTERSECTION, JOHNSON, DENNIS and SIMPSON, respectively (10), (34), (35) and (36) in Table 1. Though dramatically decreasing the size of the feature vector, they maintain a sufficient discriminative power of extracted features.

The second experiment compared the proposed approach with LBP and with Uniform LBP. Table 2 shows the results, and reports the number of bins used by the different variations of LBP with the corresponding values of EER and RR,

Table 2. Performance of NN with different strategies to reduce the set of LBP codes.

16×16 pixels			
LBP code type	Histogram bins	EER	RR
Standard	256	0.089	0.840
Uniform	59	0.091	0.842
Entropy (29)	80	0.081	0.833
Entropy (60)	41	0.095	0.827
24×24 pixels			
LBP code type	Histogram bins	EER	RR
Standard	256	0.100	0.803
Uniform	59	0.097	0.805
Entropy (29)	80	0.092	0.798
Entropy (60)	41	0.094	0.792
32×32 pixels			
LBP code type	Histogram bins	EER	RR
Standard	256	0.093	0.842
Uniform	59	0.092	0.845
Entropy (29)	80	0.085	0.840
Entropy (60)	41	0.082	0.810
36×36 pixels			
LBP code type	Histogram bins	EER	RR
Standard	256	0.113	0.746
Uniform	59	0.116	0.739
Entropy (29)	80	0.114	0.696
Entropy (60)	41	0.104	0.671

Table 3. The full set of similarity (S) and distance (D) measures.

S\D	Name	OTU	
S	JACCARD or TANIMOTO	$a/(a + b + c)$	(1)
S	DICE or CZEKANOWSKI or NEI & LI	$2 * a / (2 * a + b + c)$	(2)
S	3W-JACCARD	$3 * a / (3 * a + b + c)$	(3)
S	SOKAL & SNEATH-I	$a / (a + 2 * b + 2 * c)$	(4)
S	SOKAL & MICHENER	$(a + d) / (a + b + c + d)$	(5)
S	SOKAL & SNEATH-II	$2 * (a + d) / (2 * a + b + c + 2 * d)$	(6)
S	ROGER & TANIMOTO	$(a + d) / (a + 2 * (b + c) + d)$	(7)
S	FAITH	$(a + 0.5 * d) / (a + b + c + d)$	(8)
S	GOWER & LEGENDRE	$(a + d) / (a + 0.5 * (b + c) + d)$	(9)
S	INTERSECTION	a	(10)
S	INNERPRODUCT	$a + d$	(11)
S	RUSSELL & RAO	$a / (a + b + c + d)$	(12)
D	HAMMING	$b + c$	(13)
D	EUCLID	$\sqrt{b + c}$	(14)
D	CANBERRA or MANHATTAN or CITYBLOCK or MINKOWSKI	$b + c$	(15)
D	MEAN MANHATTAN	$(b + c) / (a + b + c + d)$	(16)
D	VARI	$(b + c) / (4 * (a + b + c + d))$	(17)
D	SIZEDIFFERENCE	$(b + c)^2 / (a + b + c + d)^2$	(18)
D	SHAPEDIFFERENCE	$(n * (b + c) - (b - c)^2) / (a + b + c + d)^2$	(19)
D	PATTERNDIFFERENCE	$4 * b * c / (a + b + c + d)^2$	(20)
D	LANCE & WILLIAMS or BRAY & CURTIS	$(b + c) / (2 * a + b + c)$	(21)
D	HELLINGER or CHORD	$2 * \sqrt{1 - a / \sqrt{(a + b) * (a + c)}}$	(22)
S	COSINE	$a / (a + b) * (a + c)$	(23)
S	GILBERT & WELLS	$\log(a) - \log(n) - \log((a + b) / n) - \log((a + c) / n)$	(24)
S	OCHIAI-I or OTSUKA	$a / \sqrt{(a + b) * (a + c)}$	(25)
S	FORBES-I	$n * a / (a + b) * (a + c)$	(26)
S	FOSSUM	$n * (a - 0.5)^2 / (a + b) * (a + c)$	(27)
S	SORGENFREI	$a^2 / (a + b) * (a + c)$	(28)
S	MOUNTFORD	$a / (0.5 * (a * b + a * c) + b * c)$	(29)
S	MCCONNAUGHEY	$(a^2 - b * c) / (a + b) * (a + c)$	(30)
S	TARWID	$(n * a - (a + b) * (a + c)) / (n * a + (a + b) * (a + c))$	(31)
S	KULCZYNSKI II	$(a / 2 * (2 * a + b + c)) / ((a + b) * (a + c))$	(32)
S	DRIVER & KROEBER	$(a / 2 * (1 / (a + b) + 1 / (a + c)))$	(33)
S	JOHNSON	$a / (a + b) + a / (a + c)$	(34)
S	DENNIS	$(a * d - b * c) / \sqrt{n * (a + b) * (a + c)}$	(35)
S	SIMPSON	$a / \min(a + b, a + c)$	(36)
S	BRAUN & BANQUET	$a / \max(a + b, a + c)$	(37)
S	FAGER & MCGOWAN	$a / \sqrt{(a + b) * (a + c)} - \max(a + b, a + c) / 2$	(38)
S	FORBES-II	$(n * a - (a + b) * (a + c)) / (n * \min(a + b, a + c) - (a + b) * (a + c))$	(39)
S	SOKAL & SNEATH-IV	$1 / 4 * (a / (a + b) + a / (a + c) + d / (b + d) + d / (c + d))$	(40)
S	GOWER	$(a + d) / \sqrt{(a + b) * (a + c) * (b + d) * (c + d)}$	(41)
S	PEARSON-I	χ^2 with $\chi^2 = n * (ad - bc)^2 / (a + b)(a + c)(c + d)(b + d)$	(42)
S	PEARSON-II	$\sqrt{\chi^2 / (n + \chi^2)}$	(43)
S	PEARSON-III	$\sqrt{\rho / (n + \rho)}$ with $\rho = (ad - bc) / \sqrt{(a + b)(a + c)(c + d)(b + d)}$	(44)
S	PEARSON & HERON-I	ρ	(45)
S	PEARSON & HERON-II	$\cos(\pi * \sqrt{b * c} / (\sqrt{a * d} + \sqrt{b * c}))$	(46)
S	SOKAL & SNEATH-III	$(a + d) / (b + c)$	(47)
S	SOKAL & SNEATH-V or OCHIAI-II	$a * d / \sqrt{(a + b) * (a + c) * (b + d) * (c + d)}$	(48)
S	COLE	$\sqrt{2 * (ad - bc) / \sqrt{a * d - b * c}^2 - (a + b) * (a + c) * (b + d) * (c + d)}}$	(49)
S	STILES	$\log_{10}(n * (ad - bc - n / 2)^2 / ((a + b) * (a + c) * (b + d) * (c + d)))$	(50)
S	YULE Q	$(a * d - b * c) / (a * d + b * c)$	(51)
D	YULE Q	$2bc / (a * d + b * c)$	(52)
S	YULE w	$(\sqrt{a * d} - \sqrt{b * c}) / (\sqrt{a * d} + \sqrt{b * c})$	(53)
S	KULCZYNSKI-I	$a / (b + c)$	(54)
S	DISPERSON	$(a * d - b * c) / (a * d + b * c)^2$	(55)
S	HAMANN	$((a + d) - (b + c)) / (a + b + c + d)$	(56)
S	MICHAEL	$4 * (a * d - b * c) / ((a + d)^2 + (b + c)^2)$	(57)
S	GOODMAN & KRUSKAL	$(\sigma - \sigma') / (2 * n - \sigma')$ with $\sigma = \max(a, b) + \max(c, d) + \max(a, c) + \max(b, d)$ and $\sigma' = \max(a + c, b + d) + \max(a + b, c + d)$	(58)
S	ANDERBERG	$(\sigma - \sigma') / (2 * n)$	(59)
S	BARONI-URBANI & BUSER-I	$(\sqrt{a * d} + a) / (\sqrt{a * d} + a + b + c)$	(60)
S	BARONI-URBANI & BUSER-II	$(\sqrt{a * d} + a - (b + c)) / (\sqrt{a * d} + a + b + c)$	(61)
S	PERCE	$(a * b + b * c) / (a * b + 2 * b * c + c * d)$	(62)
S	EYRAUD	$n^2 * (n * a - (a + b) * (a + c)) / ((a + b) * (a + c) * (b + d) * (c + d))$	(63)
S	TARANTULA	$a * (c + d) / (c * (a + b))$	(64)
S	AMPLE	$ a * (c + d) / (c * (a + b)) $	(65)

having chosen the most representative measures according to the results of the first experiment. We can observe that using MOUNTFORD measure, i.e. (29) in Table 1, and carrying out our clustering/selection procedure, we obtain an LBP coding able to achieve better performance than Uniform LBP at the expense of a higher number of bins. On the contrary, BARONI-URBANI & BUSER-I is able to finally produce a lower number of bins, with a decrease of performance that might be negligible according to the accuracy requirements. In summary, it is possible to improve the performance over Uniform codes either in terms of feature vector length or in terms of EER.

5 Conclusions

This paper presented a novel approach to the selection of the most representative LBP codes in order to obtain smaller though sufficiently discriminative feature vectors. The proposed method neither performs an unaffordable exhaustive search nor relies on codes with special patterns. It rather exploits a clustering procedure based on a measure of representativeness of the different codes. Such measure is based in turn on a suitable similarity measure among binary codes. The obtained results show that it is possible to reduce the number of LBP codes to use in building feature vectors, without affecting the classification performance too much. The experiments aimed at analyzing 65 different similarity/distance measures. Though some common aspects of behavior were detected, some measures resulted better able to improve the selection of an appropriate subset of codes, by either reducing the size of the feature vectors without a dramatic decrease in performance, or obtaining a slightly better result than Uniform LBP at the expense of using some more codes. Our future work will be focused on testing the generality of these outcomes on different classes of images. In practice, LBP can be used in many applications based on texture analysis, and it will be interesting to evaluate our approach in a different context. In particular, it will be interesting to investigate if the same similarity measures produce equivalent results on different classes of images.

References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(12), 2037–2041 (2006)
2. Chai, Z., Sun, Z., Tan, T., Mendez-Vazquez, H.: Local salient patterns - a novel local descriptor for face recognition. In: *International Conference on Biometrics (ICB)* (2013)
3. Choi, S.S., Cha, S.H., Tappert, C.C.: A survey of binary similarity and distance measures. *J. Syst. Cybern. Inf.* **8**(1), 43–48 (2010)
4. De Marsico, M., Nappi, M., Riccio, D.: ES-RU: an entropy based rule to select representative templates in face surveillance. *Multimedia Tools Appl.* **73**(1), 109–128 (2014)

5. De Marsico, M., Nappi, M., Riccio, D.: Entropy-based automatic segmentation and extraction of tumors from brain MRI images. In: Azzopardi, G., Petkov, N. (eds.) CAIP 2015. LNCS, vol. 9257, pp. 195–206. Springer, Cham (2015). doi:[10.1007/978-3-319-23117-4_17](https://doi.org/10.1007/978-3-319-23117-4_17)
6. De Marsico, M., Nappi, M., Riccio, D., Tortora, G.: Entropy-based template analysis in face biometric identification systems. *SIViP* **7**(3), 493–505 (2013)
7. Dunn, G., Everitt, B.S.: *An Introduction to Mathematical Taxonomy*. Cambridge University Press, Cambridge (1982)
8. Nappi, M., Riccio, D., De Marsico, M.: Entropy based biometric template clustering. In: ICPRAM, pp. 560–563 (2013)
9. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. *Pattern Recogn.* **29**(1), 51–59 (1996)
10. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(7), 971–987 (2002)
11. Pietikäinen, M., Hadid, A., Zhao, G., Ahonen, T.: *Computer Vision Using Local Binary Patterns*. Springer, London (2011). doi:[10.1007/978-0-85729-748-8](https://doi.org/10.1007/978-0-85729-748-8)
12. Riccio, D., Tortora, G., De Marsico, M., Wechsler, H.: EGA - ethnicity, gender and age, a pre-annotated face database. In: 2012 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications (BIOMS), pp. 1–8 (2012)
13. Riccio, D., De Marsico, M., Plasencia-Calaña, Y., Mendez-Vazquez, H.: GETSEL: gallery entropy for template selection on large datasets. In: International Joint Conference on Biometrics (IJCB 2014) (2014)
14. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: a comprehensive study. *Image Vis. Comput.* **27**(6), 803–816 (2009)
15. Sun, N., Zheng, W., Sun, C., Zou, C., Zhao, L.: Gender classification based on boosting local binary pattern. In: Wang, J., Yi, Z., Zurada, J.M., Lu, B.-L., Yin, H. (eds.) ISNN 2006. LNCS, vol. 3972, pp. 194–201. Springer, Heidelberg (2006). doi:[10.1007/11760023_29](https://doi.org/10.1007/11760023_29)
16. Topi, M., Timo, O., Matti, P., Maricor, S.: Robust texture classification by subsets of local binary patterns. In: Proceedings of 15th International Conference on Pattern Recognition, Barcelona, Spain, vol. 3, pp. 935–938 (2000)
17. Tapia, J.E., Perez, C.A., Bowyer, K.W.: Gender classification from iris images using fusion of uniform local binary patterns. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8926, pp. 751–763. Springer, Cham (2015). doi:[10.1007/978-3-319-16181-5_57](https://doi.org/10.1007/978-3-319-16181-5_57)
18. Wang, L., He, D.C.: Texture classification using texture spectrum. *Pattern Recogn.* **23**(8), 905–910 (1990)
19. Yang, Z., Ai, H.: Demographic classification with local binary patterns. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 464–473. Springer, Heidelberg (2007). doi:[10.1007/978-3-540-74549-5_49](https://doi.org/10.1007/978-3-540-74549-5_49)