Local Higher-Order Statistics (LHS) for Texture Categorization and Facial Analysis

Gaurav Sharma, Sibt Ul Hussain, Frédéric Jurie

Local Higher-Order Statistics (LHS) for Texture Categorization and Facial Analysis

Gaurav Sharma[1,2], Sibt ul Hussain[1], Frédéric Jurie[1]
gaurav.sharma@unicaen.fr, sibt.ul.hussain@gmail.com, frederic.jurie@unicaen.fr

[1]GREYC, CNRS UMR 6072, Université de Caen
[2]LEAR, INRIA Grenoble Rhône-Alpes

Abstract. This paper proposes a new image representation for texture categorization and facial analysis, relying on the use of higher-order local differential statistics as features. In contrast with models based on the global structure of textures and faces, it has been shown recently that small local pixel pattern distributions can be highly discriminative. Motivated by such works, the proposed model employs higher-order statistics of local non-binarized pixel patterns for the image description. Hence, in addition to being remarkably simple, it requires neither any user specified quantization of the space (of pixel patterns) nor any heuristics for discarding low occupancy volumes of the space. This leads to a more expressive representation which, when combined with discriminative SVM classifier, consistently achieves state-of-the-art performance on challenging texture and facial analysis datasets outperforming contemporary methods (with similar powerful classifiers).

1 Introduction

Visual categorization under multiple sources of variations *e.g.* illumination, scale, pose, *etc.*, is a challenging open problem in computer vision. Although the community has spent a lot of effort on object-category classification and object segmentation tasks [1] – leading to very powerful intermediate representation of images such as the BoW model [2,3], texture recognition has received relatively less attention despite its importance for several computer vision tasks. Texture recognition is beneficial for many applications such as mobile robot navigation or biomedical image processing. Texture analysis is also related to facial analysis *e.g.* facial expression categorization and face verification as the models developed for texture recognition can, in general, be used successfully for face analysis. Such tasks, similarly, find important applications in human computer interaction and in security and surveillance applications. This paper aims to catch up on this topic by proposing a new model providing a powerful texture representation.

Earlier works on texture analysis were focused on the development and application of filter banks *e.g.* [2, 4, 5]. They computed filter response coefficients for a number of filters or wavelets and learned their distributions. However, later works disproved the necessity of such ensembles of filters *e.g.* Ojala *et al.* [6] and Varma and Zisserman [7] showed that it is possible to discriminate between

textures using pixel neighbourhoods as small as 3×3 pixels. They demonstrated that despite the global structure of the textures, very good discrimination could be achieved by exploiting the distributions of such pixel neighbourhoods. More recently, exploiting such *micro-structures* in textures by representing images with distributions of local descriptors has gained much attention and has led to state-of-the-art performances [8–11]. However, as we discuss later, these methods suffer from several important limitations, such as the use of fixed quantization of the feature space as well as the use of heuristics to prune volumes in the feature space. In addition, they represent feature distributions with histograms and hence are restricted to the use of low order statistics.

In contrast to these previous works, we propose a model that represents images with higher order statistics of local pixel neighbourhoods. We obtain a data driven partition of the feature space using parametric mixture models, to represent the distribution of the vectors, and learn the parameters from the training data. Hence, the coding of vectors is intrinsically adapted to any classification task and the computations involved remain very simple despite the strengths. We validate our approach by extensive experiments on four challenging datasets: (i) Brodatz 32 texture dataset [12,13], (ii) KTH TIPS 2a materials dataset [14], (iii) Japanese female facial expressions dataset [15], and (iv) Labeled Faces in the Wild (LFW) dataset [16], and show that using higher-order statistics gives a more expressive description and leads to state-of-the-art performance.

1.1 Related works

Most of the earlier works on texture analysis focused on the development of filter banks and on characterizing the statistical distributions of their responses e.g. [2, 4, 5], until Ojala *et al.* [6] and, more recently, Varma and Zisserman [7] showed that statistics of small pixel neighbourhoods are capable of achieving high discrimination. Since then many methods working with local pixel neighbourhoods have been used successfully in texture and face analysis, *e.g.* [10, 11, 17].

Local pixel pattern operators, such as Local Binary Patterns (LBP) by Ojala *et al.* [6], have been very successful for local pixel neighbourhood description. LBP based image representation aims to capture the joint distribution of local pixel intensities. LBP approximates the distribution by first taking the differences between the center pixel and its neighbours and then considering just the signs of the differences. The first approximation lends invariance to gray-scale shifts and the second to intensity scaling. Local Ternary Patterns (LTP) were introduced by Tan and Triggs [10] to add resistance to noise. LTP requires a parameter t, which defines a tolerance for similarity between different gray intensities, allowing for robustness to noise. Doing so lends an important strength: LTPs are capable of encoding pixel similarity information modulo noise. However, LTP (and LBP) coding is still limited due to its hard and fixed quantization. In addition, both LBP and LTP representations usually use the so-called *uniform* patterns: patterns with at most one 0-1 and at most one 1-0 transition, when seen as circular bit strings. The use of these patterns is motivated by the empirical observation that uniform patterns account for nearly 90 percent of all

observed patterns in textures. Although it works quite well in practice, still it is a heuristic for discarding low occupancy volumes in feature space.

Most of the other recent methods, driven by the success of earlier texton based texture classification method [2] and recent advances in the field of object category classification, adopt bag-of-words models to represent textures as distributions of local textons [7, 17–24]. They learn a dictionary of textons obtained by clustering vectors (*e.g.* based on either pixel intensities, sampled on local neighbourhoods, or their differences), and then represent the image as histograms over the learnt codebook vector assignments. The local vectors are derived in multiple ways, incorporating different invariances like rotation, view point *etc.* *E.g.* [18, 19] generate an image specific texton representation from rotation and scale invariant descriptors and compare them using Earth Movers distance, whereas [6, 7, 17, 20] use a dictionary learned over the complete dataset to represent each image as histogram over this dictionary.

The motivations for this paper follow the conclusions that can be drawn from these related works. (i) As shown by [6,7], and by all the recent papers that build on these, modeling distributions of small pixel neighbourhoods (as small as 3×3 pixels) can be very effective. (ii) Unfortunately, all the previously mentioned approaches involve coarse approximations that prevent them from getting all the benefits of an accurate representation of such small neighbourhoods, and (iii) all these methods use low-order statistics while using high-order moments can give a more expressive representation. Addressing these limitations by accurately describing small neighbourhoods with their higher-order statistics, without coarse approximations, is the major contribution of the present paper.

2 The Local Higher-order Statistics (LHS) Model

As explained before, the proposed Local Higher-order Statistics (LHS) model intends to represent images by exploiting, as well as possible, the distribution of local pixel neighbourhoods. Thus, we start with small pixel neighbourhoods of 3×3 pixels and model the statistics of their local differential vectors.

Local differential vectors. We work with all possible 3×3 neighbourhoods in the image, *i.e.* $\{v^n = (v_c, v_1, \ldots, v_8)\}$ where v_c is the intensity of the center pixel and the rest are those of its 8-neighbours. We are interested in exploiting the distribution $p(v^n|I)$ of the these vectors, for a given image, to represent the image. We obtain invariance to monotonic changes in gray levels by subtracting the value of the center pixel from the rest and using the difference vector *i.e.*

$$p(v^n|I) \approx p(v|I) \;\text{ where, }\; v = (v_1 - v_c, \ldots, v_8 - v_c). \tag{1}$$

We call the vectors $\{v\}$ thus obtained as the differential vectors.

Higher order statistics. The key contribution of LHS is to use the statistics of the differential vectors $\{v|v \in I\}$ to characterize the images. Instead of using a hard and/or predefined quantization, we use parametric Gaussian mixture model (GMM) to derive a probabilistic representation of the differential space.

Defining such soft quantization, which can equivalently be seen as a generative model on the differential vectors, allows us to use a characterization method which exploits higher order statistics. We use the *Fisher score* method (Jaakkola and Haussler [25]), where given a parametric generative model, a vector can be characterized by the gradient with respect to the parameters of the model. The Fisher score, for an observed vector v *wrt.* a distribution $p(v|\lambda)$, where λ is parameter vector, is given as,

$$g(\lambda, v) = \nabla_\lambda \log p(v|\lambda). \tag{2}$$

The Fisher score, thus, is a vector of same dimensions as the parameter vector λ. For a mixture of Gaussian distribution i.e.

$$p(v|\lambda) = \sum_{c=1}^{N_k} \alpha_k \mathcal{N}(v|\mu_k, \Sigma_k) \tag{3}$$

$$\mathcal{N}(v|\mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_k|}} \exp\left\{-\frac{1}{2}(v-\mu_k)\Sigma_k^{-1}(v-\mu_k)\right\}, \tag{4}$$

the Fisher scores can be computed using the following partial derivatives (we assume diagonal Σ to decrease the number of parameters to be learnt)

$$\frac{\partial \log p(v|\lambda)}{\partial \mu_k} = \gamma_k \Sigma_k^{-1}(v-\mu_k) \tag{5a}$$

$$\frac{\partial \log p(v|\lambda)}{\partial \Sigma_k^{-1}} = \frac{\gamma_k}{2}\left(\Sigma_k - (v-\mu_k)^2\right) \tag{5b}$$

$$\text{where,} \quad \gamma_k = \frac{\alpha_k p(v|\mu_k, \Sigma_k)}{\sum_k \alpha_k p(v|\mu_k, \Sigma_k)} \tag{5c}$$

where the square of a vector is element-wise one. In the derivatives above we can see that the information based on the first and second powers of the differential vectors are also coded; these are higher order statistics for the differential vectors. After obtaining the differential vectors corresponding to every pixel neighbourhood in the image, we compute the image representation as the average vector over all of them. We normalize each dimension of the image vector to zero mean and unit variance. To perform the normalization we use training vectors and compute multiplicative and additive constants to perform whitening per dimension [26]. We also incorporate two normalizations (on image vector x) [27] i.e. power normalization,

$$(x_1, \ldots, x_d) \leftarrow (\text{sign}(x_1)\sqrt{|x_1|}, \ldots, \text{sign}(x_d)\sqrt{|x_d|}), \tag{6}$$

and L2 normalization,

$$(x_1, \ldots, x_d) \leftarrow \left(\frac{x_1}{\sqrt{\sum x_i^2}}, \ldots, \frac{x_d}{\sqrt{\sum x_i^2}}\right). \tag{7}$$

Algorithm 1 Computing Local higher-order statistics (LHS)

1: Randomly sample 3×3 pixels differential vectors $\{v \in I | I \in \mathcal{I}_{train}\}$
2: Learn the GMM parameters $\{\alpha_k, \mu_k, \Sigma_k | k = 1 \dots K\}$ with EM algorithm on $\{v\}$
3: Compute the higher-order Fisher scores for $\{v\}$ using equations (5)
4: Compute means C_μ^d and variances C_Σ^d for each dimension d
5: **for all** images $\{I\}$ **do**
6: Compute all differential vectors $v \in I$
7: Compute the Fisher scores for all features $\{v\}$ using equations (5)
8: Compute the image representation x as the average score over all features
9: Normalize each dimension d as $x^d \leftarrow (x^d - C_\mu^d)/C_\Sigma^d$
10: Apply normalizations, equations (6) and (7)
11: **end for**

The whole algorithm, which is remarkably simple, is summarized in Alg. 1. Finally, we use the vectors obtained as the representation of the images and employ a discriminative linear support vector machine (SVM) as the classifier in a supervised learning setup.

Relation to LBP/LTP. We can view LHS vectors as generalization of local binary/ternary patterns (LBP/LTP) [6, 10]. In LBP every pixel is coded as a binary vector of 8 bits with each bit indicating whether the current pixel is of greater intensity when compared to (one of the 8) its neighbours. We can derive the LBP [6] by thresholding each coordinate of our differential vectors at zero. Hence the LBP space can be seen as a discretization of the differential space into two bins per coordinate. Similarly, we can discretize the differential space into more number of bins, with three bins per coordinate i.e. $(-\infty, -t), [-t, t], (t, -\infty)$ we arrive at the local ternary patterns [10] and so on. The use of *uniform patterns* (patterns with exactly one 0-1 and one 1-0 transitions), in both LBP/LTP, can be only seen as an empirically derived heuristic for ignoring volumes in differential space which have low occupancies. Thus, the binary/ternary patterns are obtained with a quantization step and rejection heuristic while in our case similar information is learnt from data.

3 Experimental Validation

The experimental validation is done on four challenging publicly available datasets of textures and faces. We first discuss implementation details then present the datasets and finally give the experimental results for each dataset.

As our focus is on the rich and expressive representation of local neighbourhoods, we use a standard classification framework based on linear SVM. As linear SVM works directly in the input feature space, any improvement in the performance is directly related to a better encoding of local regions, and thus helps us gauge the quality of our features.

Implementation details. We use only the intensity information of the images and convert color images, if any, to grayscale. We consider two neighbourhood

sampling strategies (i) rectangular sampling, where the 8 neighbouring pixels are used, and (ii) circular sampling, where, like in LBP/LTP [6, 10], we interpolate the diagonal samples to lie on a circle, of radius one, using bilinear interpolation. We randomly sample at most 500,000 features from training images to learn Gaussian mixture model of the vectors, using the EM algorithm initialized with k-means clustering. We keep the number of components as an experimental parameter (Sec. 3.3). We also use these features to compute the normalization constants, by first computing their Fisher score vectors and then computing (per coordinate) mean and variance of those vectors (Alg. 1). We use the average of all the features from the image as the representation for the image. However, for the facial expression dataset we first compute the average vectors for non overlapping cells of 10×10 pixels and concatenate these for all cells to obtain the final image representation. Such gridding helps in capturing spatial information in the image and is standard in face analysis [28, 29]. We crop the 256×256 face images to a ROI of $(66, 96, 186, 226)$, to focus on the face, before feature extraction and do not apply any other pre-processing. Finally, we use linear SVM as the classifier with the cost parameter C set using five fold cross validation on the current training set.

Baselines. We consider baselines of single scale LBP/LTP features generated using the same samplings as our LHS features. We use histogram representation over uniform LBP/LTP features. We L1 normalize the histograms and take their square roots and use them with linear SVM. It has been shown that taking square root of histograms transforms them to a space where the dot product corresponds to the non linear Bhattacharyya kernel in the original space [30]. Thus using linear SVM with square root of histograms is equivalent to SVM with non linear Bhattacharyya kernel. Hence, our baselines are strong baselines.

3.1 Texture categorization

Brodatz – 32 Textures dataset[1] [12, 13] is a standard dataset for texture recognition. It contains 32 texture classes *e.g.* bark, beach-sand, water, with 16 images per class. Each of the image is used to generate 3 more images by (i) rotating, (ii) scaling and (iii) both rotating and scaling the original image – note that Brodatz-32 [12] is a more challenging dataset than original Brodatz and includes both rotation and scale changes. The images are 64×64 pixels histogram normalized grayscale images. We use the standard protocol [11], of randomly splitting the dataset into two halves for training and testing, and report average performance over 10 random splits.

KTH TIPS 2a dataset[2] [14] is a dataset for material categorization. It contains 11 materials *e.g.* cork, wool, linen, with images of 4 samples for each material. The samples were photographed at 9 scales, 3 poses and 4 different illumination conditions. All these variations make it an extremely challenging dataset. We

[1] http://www.cse.oulu.fi/CMV/TextureClassification
[2] http://www.nada.kth.se/cvap/datasets/kth-tips/

Table 1. Results (avg. accuracy and std. dev.) on the different datasets.

(a) Rectangular sampling (8-pixel neighbourhood)

	Brodatz–32	KTH TIPS 2a	JAFFE E1	JAFFE E2
LBP baseline	87.2 ± 1.5	69.8 ± 6.9	86.9 ± 2.6	56.5 ± 21.0
LTP baseline	95.0 ± 0.8	69.3 ± 5.3	93.6 ± 1.8	57.2 ± 16.3
LHS (ours)	**99.3 ± 0.3**	**71.7 ± 5.7**	**95.6 ± 1.7**	**64.6 ± 19.2**

(b) Circular sampling (bilinear interpolation for diag. neighbours)

	Brodatz–32	KTH TIPS 2a	JAFFE E1	JAFFE E2
LBP baseline	87.3 ± 1.5	69.8 ± 6.7	94.3 ± 2.1	61.8 ± 24.1
LTP baseline	94.9 ± 0.8	71.3 ± 6.3	95.1 ± 1.8	60.6 ± 20.8
LHS (ours)	**99.5 ± 0.2**	**73.0 ± 4.7**	**96.3 ± 1.5**	**63.2 ± 16.5**

use the standard protocol [11, 14] and report the average performance over the 4 runs, where every time all images of one sample are taken for test while the images of the remaining 3 samples are used for training.

Tab. 1 (col. 1 and 2) shows the results for the different methods on these texture datasets. We achieve a near perfect accuracy of 99.5% on the Brodatz dataset. Our best method outperforms the best LBP and LTP baselines by 12.2% and 4.5% respectively and demonstrates the advantage of using rich, higher-order, data-adaptive encoding of local neighbourhoods compared to fixed quantization based LBP and LTP representations. Brodatz dataset contain texture images with scale and rotation variations, hence, the high accuracy achieved on the dataset leads us to conclude that texture recognition can be done almost perfectly under the presence of rotation and scaling variations.

On the more challenging KTH TIPS 2a dataset, the best performance is far from saturated at 73%. The gain in accuracy over LBP and LTP is 3.2% and 1.7% respectively. This dataset has much stronger variations in scale, illumination conditions, pose, *etc.*, than the Brodatz dataset and the experiment is of texture categorization of unseen sample *i.e.* the test images are of a sample not seen in training. Our descriptor again outperforms LBP/LTP and demonstrates its higher discrimination power and the generalization capability.

3.2 Facial analysis

Japanese Female Facial Expressions (JAFFE)[3] [15] is a dataset for facial expression recognition. It contains 10 different females expressing 7 different emotions *e.g.* sad, happy, angry. We perform expression recognition for both known persons, like earlier works [31], and for unknown person. In the first (experiment E1), one image per expression for each person is used for testing while remaining ones and used for training. Thus, the person being tested is

[3] http://www.kasrl.org/jaffe.html

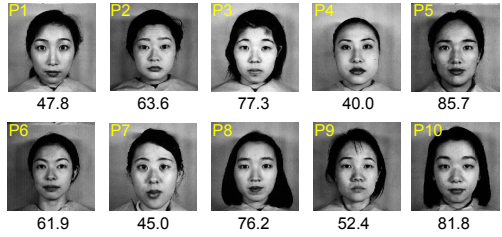

Fig. 1. The images of the 10 persons in the neutral expression. The number below is the categorization accuracy for all 7 expressions for the person (see Sec. 3.2).

present (different image) in training. In the second (experiment E2), all images of one person are held out for testing while the rest are used for training. Hence, there are no images of the person being tested in the training images, making the task more challenging. For both cases, we report the mean and standard deviation of average accuracies of 10 runs.

Tab. 1 (col. 3 and 4) shows the performance of the different methods. On the first experiment (E1) we obtain very high accuracies as the task is of recognition of expressions, from a never seen image, of a person present in the training set. Our method again outperforms LBP and LTP based representation by 2% and 1.2% respectively. On the more challenging second experiment (E2) we see that the accuracies are much less than E1. Our best accuracy is again better than the best LBP and LTP accuracies by 2.8% and 4% respectively. Fig. 1 shows one image of each of the 10 persons in the dataset along with the expression recognition accuracy for that person. We can see the very high intra-person differences in this dataset, which results in very different accuracies for the different persons and hence high standard deviation, for all the methods.

Labeled Faces in Wild (LFW) [16] is a popular dataset for face verification by unconstrained pair matching *i.e.* given two real-world face images decide whether they are of the same person or not. LFW contains 13,233 face images of 5749 different individuals of different ethnicity, gender, age, *etc.* It is an extremely challenging dataset and contains face images with large variations in pose, lighting, clothing, hairstyles, *etc.* LFW dataset is organized into two parts: 'View 1' is used for training, validation (*e.g.* for choosing the parameters) while 'View 2' is only for final testing and benchmarking. In our setup, we follow the specified training and evaluation protocol. We use the aligned version of the faces as provided by Wolf *et al.* [32][4].

We work in the restricted unsupervised task of the LFW dataset *i.e.* (i) we use strictly the data provided without any other data from any other source and (ii) we do not utilize class labels while obtaining the image representation. We divide the 50×40 pixels resized images into 5×4 grid of 10×10 pixels cells.

[4] http://www.openu.ac.il/home/hassner/data/lfwa/

We compute the LHS representations for each cell separately and compute the similarity between image pairs as the mean of L2 distances between the representations of corresponding cells. We classify image pairs into same or not same by thresholding on their similarity. We choose the testing threshold as the one which gives the best classification accuracy on the training data. We obtain an accuracy of 73.4% with a standard error on the mean of 0.4%. This is the highest performance till date in the unsupervised setting for the dataset. We compare with other approaches, including those based on LBP in Sec. 3.4.

3.3 Effect of sampling and number of components

Tab. 1 gives the results with (a) rectangular 3×3 pixel neighbourhood and (b) LBP/LTP like circular sampling of 8 neighbours, where the diagonal neighbour values are obtained by bilinear interpolation. Performance on the Brodatz dataset is similar for both the samplings while that for KTH and JAFFE datasets differ. In general, the circular sampling seems to be better for all the methods. We note that the variations and difficulty of Brodatz dataset are much less than the other two datasets and hence is possibly well represented by either of the two samplings. Thus, we conclude that, in general, circular sampling is to be preferred as it seems to generate more discriminative statistics.

Fig. 2 shows the performance on the two texture datasets for different number of mixture model components. As this number increases the vector length increases proportionally. Although lower number of components lead to a compact representation, larger numbers lead to better quantization of the space and hence more discriminative representations. We observe that the performance, for both the datasets, increases with the number of components and seems to saturate after a value of 128. Hence, we report results for 128 components. For Brodatz dataset, we see that even with only 16 components the method is able to achieve more than 99% accuracy, highlighting the fewer variations in the dataset. For the KTH dataset we gain significantly by going from 16 to 128 components (6.8 points) which suggests that for more challenging tasks a more descriptive representation is beneficial.

3.4 Comparison with existing methods

Tab. 2 shows the performance of our method along with existing methods. On the Brodatz dataset we outperform all methods and to the best of our knowledge report, near perfect, state-of-the-art performance. Similarly, on the JAFFE and LFW datasets we achieve the best results reported till date.

On the KTH dataset, Chen *et al.* [11], for their recently proposed Weber law based features, report an accuracy of 64.7% with KNN classifier. Caputo *et al.* [14] report 71.0% for their 3-scale LBP and *non-linear* chi-squared RBF kernel based SVM classifier. In comparison we use linear classifiers which are not only fast to train but also need only a vector dot product at test time (*c.f.* kernel computation with support vectors which is of the order of number of training features). Note Caputo *et al.* obtain their best results with multi

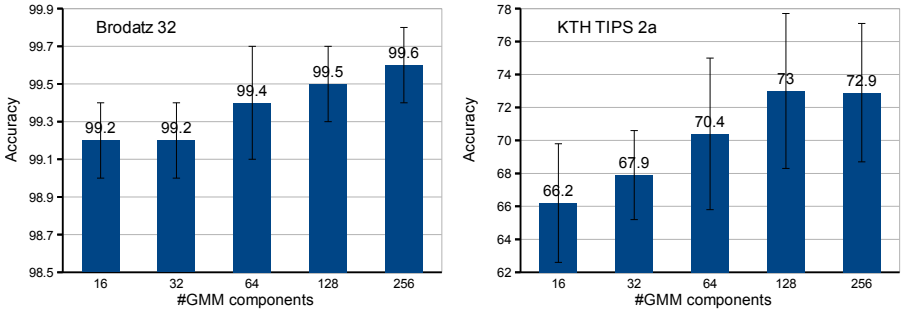

Fig. 2. The accuracies of the method for different number of GMM components for Brodatz (left) and KTH TIPS 2a (right) dataset (see Sec. 3.3)

Table 2. Comparison with current methods with comparable experimental setup (reports accuracy, see Sec. 3.4).

(a) Brodatz–32

Method	Acc.
Urbach *et al.* [33]	96.5
Chen *et al.* [11]	97.5
LHS (ours)	**99.3**

(b) KTH TIPS 2a

Method	Acc.
Chen *et al.* [11]	64.7
Caputo *et al.* [14]	71.0
LHS (ours)	**73.0**

(c) JAFFE

Method	Acc.
Shan *et al.* [29]	81.0
Feng *et al.* [28]	93.8
LHS (ours)	**95.6**

(d) LFW (aligned)

Method	Acc.
Javier *et al.* [34]	69.5 ± 0.5
Seo *et al.* [35]	72.2 ± 0.5
LHS (ours)	**73.4 ± 0.4**

scale features and a complex decision tree (with non-linear classifiers at every node). We expect our features to outperform their features with similar complex classification architecture.

Tab. 2 (d) reports accuracy rates of our method and those of competing unsupervised methods[5] on LFW dataset. Our method not only outperforms the LBP baseline (LBP with χ^2 distance) [34] by 3.9% but also gives 1.2% better performance than current state-of-the-art Locally Adaptive Regression Kernel (LARK) features of [35]. The better performance of our features, compared to the LBP baseline and fairly complex LARK features, on this difficult dataset once again underlines the fact that local neighbourhood contains a lot of discriminative information. It also demonstrates the representational power of our features which are successful in encoding the information which is missed by other methods.

[5] results reproduced from webpage: http://vis-www.cs.umass.edu/lfw/results.html

Thus the proposed method is capable of achieving state-of-the art results while being computationally simple.

4 Conclusions

We have presented a model that captures higher-order statistics of small local neighbourhoods to produce a highly discriminative representation of the images. Our experiments, on two challenging texture datasets and two challenging facial analysis datasets, validate our approach and show that the proposed model encodes more local information than the competing methods and therefore achieves state-of-the-art results.

Although we have shown that features based on local neighbourhoods can give very good results by themselves, still combining them with more global features would be a promising direction which we will explore in future.

Aknowledgement. This work was funded by the ANR, grant reference ANR-08-SECU-008-01/SCARFACE.

References

1. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html (2010)
2. Leung, T.J., Malik, J.: Representing and recognizing the visual appearance of materials using three-dimensional textons. IJCV **43** (2001) 29–44
3. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: ICCV. (2003)
4. Cula, O.G., Dana, K.J.: Compact representation of bidirectional texture functions. In: CVPR. (2001)
5. Zhu, S.C., Wu, Y., Mumford, D.: Filters, random-fields and maximum-entropy (FRAME): Towards a unified theory for texture modeling. IJCV **27** (1998) 107–126
6. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. PAMI **24** (2002) 971–987
7. Varma, M., Zisserman, A.: Texture classification: Are filter banks necessary? In: CVPR. (2003)
8. Pietikinen, M., Hadid, A., Zhao, G., Ahonen, T.: Computer Vision Using Local Binary Patterns. Springer (2011)
9. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: Application to face recognition. PAMI **28** (2006)
10. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. TIP **19** (2010) 1635–1650
11. Chen, J., Shan, S., He, C., Zhao, G., Pietikainen, M., Chen, X., Gao, W.: WLD: A robust local image descriptor. PAMI **32** (2010) 1705–1720
12. Valkealahti, K., Oja, E.: Reduced multidimensional co-occurence histograms in texture classification. PAMI **20** (1998) 90–94

13. Brodatz, P.: Textures: A Photographic Album for Artists and Designers. Dover Publications, New York (1966)
14. Caputo, B., Hayman, E., Mallikarjuna, P.: Class-specific material categorisation. In: ICCV. (2005)
15. Lyons, M.J., Akamatsu, S., Kamachi, M., Gyoba, J.: Coding facial expressions with gabor wavelets. In: AFGR. (1998)
16. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst (2007)
17. Liu, L., Fieguth, P., Kuang, G.: Compressed sensing for robust texture classification. In: ACCV. (2010)
18. Lazebnik, S., Schmid, C., Ponce, J.: A sparse texture representation using local affine regions. PAMI **27** (2005) 1265–1278
19. Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: A comprehensive study. IJCV **73** (2007) 213–238
20. Varma, M., Zisserman, A.: A statistical approach to texture classification from single images. IJCV **62** (2005) 61–81
21. Croiser, M., Griffin, L.D.: Using basic image features for texture classification. IJCV **88** (2010) 447–460
22. Hayman, E., Caputo, B., Fritz, M., Eklundh, J.O.: On the significance of real world conditions for material classification. In: ECCV. (2004)
23. Xu, Y., Ji, H., Fermuller, C.: View point invariant texture description using fractal analysis. IJCV **83** (2009) 85–100
24. Xu, Y., Yang, X., Ling, H., Ji, H.: A new texture descriptor using multifractal analysis in multi-orientation wavelet pyramid. In: CVPR. (2010)
25. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: NIPS. (1999)
26. Bishop, C.M.: Pattern recognition and machine learning. Springer (2006)
27. Perronnin, F., Sánchez, J., Mensink, T.: Improving the Fisher kernel for large-scale image classification. In: ECCV. (2010)
28. Feng, X., Pietikinen, M., Hadid, T.: Facial expression recognition with local binary patterns and linear programming. Pattern Recognition and Image Analysis **15** (2005) 546–548
29. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: A comprehensive study. IVC **27** (2009) 803–816
30. Vedaldi, A., Zisserman, A.: Efficient additive kernels using explicit feature maps. In: CVPR. (2010)
31. Liao, S., Fan, W., Chung, A.C., Yan Yeung, D.: Facial expression recognition using advanced local binary patterns, tsallis entropies and global appearance features. In: ICIP. (2006)
32. Wolf, L., Hassner, T., Taigman, Y.: Similarity scores based on background samples. In: ACCV. (2009)
33. Urbach, E.R., Roerdink, J.B., Wilkinson, M.H.: Connected shape-size pattern spectra for rotation and scale-invariant classification of gray-scale images. PAMI **29** (2007) 272–285
34. Javier, R.S., Rodrigo, V., Mauricio, C.: Recognition of faces in unconstrained environments: a comparative study. EURASIP Journal on Advances in Signal Processing (2009)
35. Seo, H.J., Milanfar, P.: Face verification using the LARK representation. Information Forensics and Security, IEEE Transactions on **6** (2011) 1275–1286