Kernel-mapped Histograms of Multi-scale LBPs for Tree Bark Recognition

Milan Sulc and Jiri Matas
Center for Machine Perception
Dept. of Cybernetics, Faculty of Electrical Engineering
Czech Technical University in Prague
{sulcmila,matas}@cmp.felk.cvut.cz

***Abstract*—We propose a novel method for tree bark identification by SVM classification of feature-mapped multi-scale descriptors formed by concatenated histograms of Local Binary Patterns (LBPs). A feature map approximating the histogram intersection kernel significantly improves the methods accuracy. Contrary to common practice, we use the full 256 bin LBP histogram rather than the standard 59 bin histogram of uniform LBPs and obtain superior results. Robustness to scale changes is handled by forming multiple multi-scale descriptors.**

Experiments conducted on a standard dataset show 96.5% accuracy using ten-fold cross validation. Using the standard 15 training examples per class, the proposed method achieves a recognition rate of 82.5% and significantly outperforms both the state-of-the-art automatic recognition rate of 64.2% and human experts with recognition rates of 56.6% and 77.8%. Experiments on standard texture datasets confirm that the proposed method is suitable for general texture recognition.

I. Introduction

Recognition of natural objects is a challenging computer vision problem that requires dealing with irregular shapes and textures with high variability. Interest in methods for image-based classification of specific parts of plants like leaves, flowers or bark has grown recently [1]–[4]. The application potential of plant recognition has increased as devices equipped with cameras became ubiquitous, making intelligent field guides, education tools or automation in forestry practical.

In the paper, we focus on bark recognition. As a recognition trait, bark has advantages over the more commonly used leafs: it is available all year round and it is easy to acquire images of it in a canonical way, leaving scale as the only major geometric factor to deal with. On the other hand, it is applicable for trees only and its often rough surface make the appearance sensitive to illumination in a way that is difficult to model.

The proposed bark recognition method uses multi-scale concatenated histograms of Local Binary Patterns as a descriptor. The classifier is a linear SVM with probabilistic output [5], [6] using an approximate kernel map [7] for the histogram-intersection kernel. The proposed linear feature-mapped SVM based on multi-scale LBPs has significantly outperformed the state-of-the-art in terms of accuracy and leads to a compact and fast classifier. Since the meethod is intended also for mobile devices, fast evaluation and low memory footprint are required. We also show that, unexpectedly, the common practice of using uniform LBPs leads to a decrease in accuracy.

978-1-4799-0883-7/13/$31.00 ©2013 IEEE

A. Related work

The problem of automatic tree identification based on photos of bark is usually formulated as texture recognition.

Chi et al. [8] proposed a method using Gabor filter banks. Their accuracy reaches 96% on a dataset with 8 classes, 25 images (132×132 px) per class. The dataset was not published. Wan et al. [9] performed a comparative study of bark texture features. Four approaches were tested: the gray level run-length method, co-occurrence matrices method (COMM), histogram method and auto-correlation method. COMM was found to be the best, achieving 72% recognition rate using the 1-NN classifier on an unpublished dataset of 160 images from 9 classes. The author also show that the performance of all classifiers improved significantly, up to 89% using COMM and 1-NN, when color information was added.

Song et al. [10] presented a feature-based method for bark recognition using a combination of grey-level co-occurrence matrix (GLCM) and a binary texture feature called Long Connection Length Emphasis. Using 1-NN classification, 87.8% accuracy was achieved on an unpublished dataset which contains 90 manually cropped (200×200) images of 18 classes. Huang et al. [11] used GCLM together with Fractal Dimension Features for bark description. The classification was performed by artificial neural networks on an unpublished dataset of 360 images of 24 classes. The highest recognition rate reached was 91.67%.

Since the image data used in the experiments discussed above are not available, it is difficult to assess the quality of the results and to perform comparative evaluation.

Fiel and Sablatnig [3] proposed methods for automated identification of tree species from images of the bark, leaves and needles. For bark description, SIFT features were discretized into a Bag-of-Words. A Support Vector Machine with radial basis function (RBF) kernel was used for classification.

Fiel and Sablatnig introduced the Österreichische Bundesforste AG (Austrian Federal Forests) bark dataset consisting of 1182 photos from 11 classes. This dataset will be referred to as the AFF bark dataset. A recognition accuracy of 64.2% and 69.7% was achieved for training sets with 15 and 30 images per class (where available) respectively.

Fiel and Sablatnig also describe an experiment with two human experts - a biologist and a forest ranger - both employees of Österreichische Bundesforste AG. Their classification rate on a subset of the dataset with 9 images per class, 99 images in total, was 56.6% (biologist) and 77.8% (forest ranger).

Sixta proposed a method for leaf and bark recognition in [12] using histograms of uniform LBPs on different scales (with use of multi-scale block LBP) for bark description. However, only two of the ten scales were computed through LBP extraction, the others were described using linear interpolation and extrapolation. The 1-NN classifier was used. The distance between two images was defined as the best alignment of their histograms using the χ^2 distance function. For evaluation, Sixta used an incomplete version of the AFF bark dataset, containing only 1081 images. With two-fold cross-validation, the recognition rate on this dataset was 70.1%.

B. Mobile apps for image-based tree and plant recognition

Pl@ntNet (plantnet-project.org) is an interactive plant identification and collaborative information system providing an image sharing and retrieval application for plant identification. The Pl@ntNet Tree Database combines information from images of the habitat, flower, fruit, leaf and bark. The algorithms used in Pl@ntNet are not publicly documented.

Columbia University, the University of Maryland and the Smithsonian Institution have been developing methods for visual recogntion of leaf shape since 2008 [1]. Based on this system, an electronic field guide called Leafsnap [4] was created. This guide is available as a mobile app, but image processing takes place on a server which may be limiting due to problems with connectivity.

The rest of the paper is structured as follows: In Section II, we describe the proposed texture description and classification method and discuss its advantages for practical applications. Experimental results on the AFF bark dataset and on two standard texture datasets (Brodatz32,UIUCTex) are presented in Section III. Presented results are summarized and the conclusions are drawn in Section IV.

II. The Proposed Method

A. Bark description by multi-scale Local Binary Patterns

The proposed description of bark texture is based on histograms of Local Binary Patterns [13], [14]. The LBP operator computes the signs of differences between the center pixel and its neighbours, so the description is invariant to locally monotonic illumination changes.

In [15] the general $\mathrm{LBP}_{P,R}$ operator is introduced, based on a circularly symmetric neighbourhood of P members on a circle of radius R. $\mathrm{LBP}_{8,1}$ is similar to the classical LBP operator on a 3×3 neighbourhood, except that the values of the diagonal pixels are determined by interpolation and that the pixels in the neighbour set are indexed so that they form a circular chain.

To exploit information from multiple scales, c $\mathrm{LBP}_{8,R}$ histograms are concatenated into one multi-scale descriptor. Robustness to scale changes is handled by creating n_{conc} multi-scale descriptors for one image, as in Fig. 1. For multiple

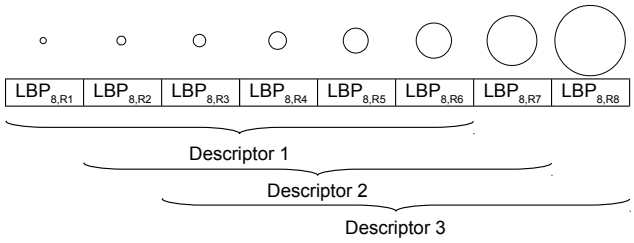
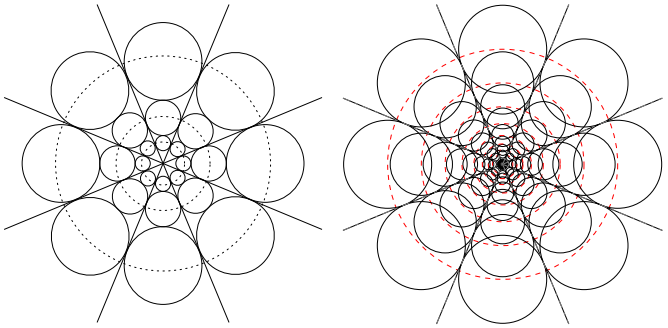


Fig. 1. The proposed histogram concatenation



(a) Scale space by Mäenpää [17] (b) The proposed scale space

Fig. 2. The effective areas of filtered pixel samples in a multi-resolution $\mathrm{LBP}_{8,R}$ operator

multi-scale descriptors per image, a scale pyramid needs to be built with a constant ratio among individual scales, i.e. an exponential scale change.

Multi-scale Block LBP [16] are a high-performance approach to compute LBPs on different scales, as the computation is done based on average values of block subregions which can be computed very efficiently using integral images. However, MB-LBP are not applicable for a dense scale space with constant scale ratio among the histograms, as the block dimensions must be divisible by 3. Therefore, a different approach has to be used. We extract $\mathrm{LBP}_{8,R}$ from circular neighbourhoods with exponentially growing radius R. Because the intensity values are obtained from single pixel positions, Gaussian filtering is used to overcome pixel noise. The combination of an exponentially growing multi-resolution LBP and Gaussian filtering is known from [17], where the radii of the LBP operators are chosen so that the effective areas of different scales touch each other, as in Fig. 2a. But as a finer scale change is needed, we chose a step of $\sqrt{2}$ between adjacent scales which is equivalent to decreasing the image area to one half.

$$R_i = R_{i-1}\sqrt{2} \tag{1}$$

We have also tried a step of 2 and $\sqrt[3]{2}$, but the change of $\sqrt{2}$ gives the best results. The proposed scale space is illustrated in Fig. 2b.

The effective areas (with radius r_i) for given scale i are chosen in such a way that they touch each other.

$$r_i = R_i \sin\frac{\pi}{8} \tag{2}$$

Similar to [17], the filters are designed so that 95% of their mass lies within the solid circles. Using the Gauss error function erf, the proportion of data mass in z standard

deviations for normal distribution equals $\mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)$. From this the standard deviation for the Gaussian filter can be simply derived:

$$\sigma_i = \frac{r_i}{z} = \frac{r_i}{\sqrt{2} \cdot \mathrm{erf}^{-1}(0.95)} \approx \frac{r_i}{1.959964} \tag{3}$$

The setting of parameter c is discussed in Section III. As the image scale is bounded by the width of the tree trunk and users intuitively understand that small regions may not be sufficient, only $n_{conc} = 3$ multi-scale descriptors per image are used.

Ojala et al. [15] have shown that the 58 uniform LBPs (i.e. patterns with at most 2 spatial transitions / bitwise 0-1 changes) often carry the key information and they shown to be a very powerful texture feature. Usually a 59 bin histogram is used - with 1 "miscellaneous" bin for all non-uniform patterns. Since the SVM is used for classification, the full 256-bin LBP can be used (contrary to common practice), as the SVM itself will assign low weights to less important histogram bins. In section III-A we will show that the full 256-bin LBP description leads to a slightly better accuracy (though the training and testing are more demanding).

B. Rotation invariance

The proposed recognition method works with the up-is-up assumption, as for users it is a requirement easy to satisfy, since the tree trunk orientation is unambiguous. However, we also propose a rotation invariant description.

A simple rotation invariant description can be achieved by using a histogram of 36 Rotation Invariant Local Binary Patterns, also named as LBP^{ri} [15] or LBP_{ROT} [18]. However, using LBP^{ri} leads to the loss of information about relative distribution of different orientations, as it normalizes each pattern individually.

$$\mathrm{LBP}^{ri}_{P,R} = \min\left\{\mathrm{ROR}\left(\mathrm{LBP}_{P,R}, i\right) \mid t = 0, 1, .., P-1\right\} \tag{4}$$

where $\mathrm{ROR}\left(x, i\right)$ performs a circular bit-wise right shift on the P-bit number x i-times.

A more discriminative rotation invariant description can be achieved with Local Binary Pattern Histogram Fourier Features [19] (LBP-HF). From a histogram of uniform LBPs (with one bin for non-uniform patterns), 38 rotation invariant features are computed using Discrete Fourier Transform (DFT). Experiments with LBP-HF description will be performed in Section III-A.

We note that LBP-HF (as well as LBP^{ri}) are rotation invariant only in the sense of a circular bit-wise shift, i.e. rotation by multiples of 45° when using $\mathrm{LBP}_{8,R}$.

C. Support Vector Machine and feature maps

A linear Support Vector Machine classifier on kernel mapped data meets the requirements for practical applicability for its fast evaluation, high accuracy and low storage load. Since we are working with histograms, a suitable non-linear kernel is the Intersection kernel K_{inters} (also Histogram intersection kernel) which is an additive kernel defined as:

$$K_{\text{inters}}(x, y) = \sum_{b=1}^{B} k_{\text{inters}}(x_b, y_b) = \sum_{b=1}^{B} \min\{x_b, y_b\} \tag{5}$$

where b denotes the histogram bin index, B is the total number of bins and k_{inters} is itself a kernel on the non-negative real numbers. We make use of the kernel approximation using an explicit feature map introduced in [7] which is shown to greatly reduce the train/test times of SVM implementations. The data vectors were expanded with a 3-dimensional approximated kernel map using the implementation in VLFeat [20] library. We also considered using the χ^2 kernel map, but Intersection provides better results.

After expanding the data vectors, a linear SVM solver from LIBOCAS library (Version 0.97) implementing OCAS (Optimized Cutting plane Algorithm for Support Vector Machines) [21] is used. This solver is designed to use the regularisation parametrisation in C, i.e. the task defined as:

$$\mathbf{w}^* = \underset{\mathbf{w} \in \mathbb{R}^n}{\operatorname{argmin}} \left[\frac{1}{2} \left\| \mathbf{w}^2 \right\| + C \sum_{i=1}^{m} \ell_i(\mathbf{x}_i, \mathbf{w}) \right] \tag{6}$$

where ℓ_i denotes the Hinge loss.

$$\ell_i = \max\left\{0, 1 - y_i \left\langle \mathbf{x}_i, \mathbf{w} \right\rangle\right\} \tag{7}$$

We set the parameter C according to the number of samples N. This is also known as parametrisation in λ which was chosen empirically as $\lambda = 10^{-6}$.

$$C = \frac{1}{\lambda N} \tag{8}$$

The multi-class classification problem is approached using the 'One versus All' scheme. To ensure SVM results comparability, sigmoid functions are used to map the SVM outputs into probabilities using the improved algorithm for Platt's probabilistic output [5] introduced by Lin et al [6].

Having the probability output, the class (and scale) with the highest posterior probability is selected as a result.

We only need to store one vector of weights and 2 scalar sigmoid parameters per class, thus the storage demands are very low. Larger number of training samples is required to train the SVM and fit the sigmoid properly. Different images should be used for SVM training and sigmoid fitting, but because we have a smaller dataset, we use the same training images for both procedures. Compared to k-NN classification, SVM suffers from higher training time , but the testing is much faster and independent on the size of the training set. And in typical applications the data are learned only once.

D. Handling different resolutions

Images of similar bark area can be obtained in different pixel resolutions, depending on the device used for image acquisition. Moreover, the use of higher resolution leads to slower image processing. Therefore, all images are resized to a normalized width, preserving the aspect ratio.

III. Experimental results

The proposed method has been evaluated on three image datasets: the AFF bark dataset [3] and two standard texture datasets - Brodatz32 [22], [23] and UIUCTex [24]. For all experiments on the AFF dataset, images were resized to a width of 1000 pixels. Since both standard texture datasets contain images of normalized size, no further normalisation was needed in this case.

Parameter settings. The number of scales c at which LBPs are calculated and histogrammed to form a single descriptor is an important parameter of the method. In a preliminary experiment on the AFF bark dataset, the recognition rate as a function of the number c was measured. In this experiment, only one descriptor per image ($n_{conc} = 1$) was used. Evaluation was carried out with ten-fold cross validation.

The results, see the blue line with error bars in Figure 3, show that the recognition accuracy grows with c. Since the memory requirements increase linearly and the computation complexity super-linearly with c, we set $c = 6$. For higher values of c, recognition accuracy is not increasing statistically significantly.

In the same experiment, we checked the benefit of kernelization. Results for a linear SVM operating directly on the concatenated histograms are shown in red in Figure 3. Kernelizing the data strongly improves the classification accuracy.

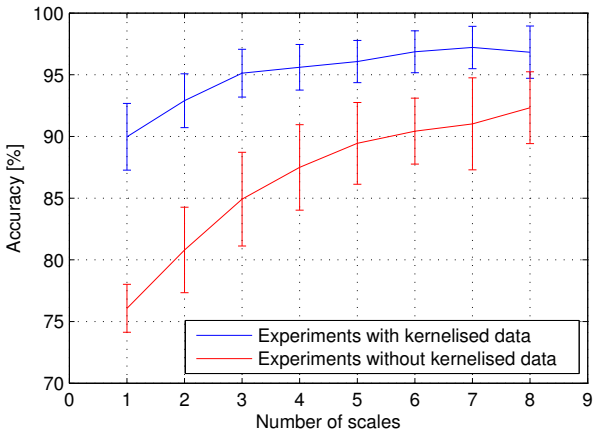


Fig. 3. Classification accuracy on the AFF bark dataset with a single multi-scale descriptor with and without kernelization (feature mapping)

A. Evaluation of the proposed method on the AFF bark dataset

The dataset collected by *Österreichische Bundesforste – Austrian Federal Forests* was introduced in 2011 by Fiel and Sablatnig [3] and contains 1182 bark images from 11 classes. The resolution of these image samples is diverse (between 0.4 Mpx and 8.0 Mpx).

TABLE I. The AFF bark dataset

#	Class	Photos	#	Class	Photos
1	Ash	33	7	Mountain oak	77
2	Beech	16	8	Scots pine	190
3	Black pine	166	9	Spruce	213
4	Fir	127	10	Swiss stone pine	96
5	Hornbeam	42	11	Sycamore maple	22
6	Larch	200			

Three variants of the proposed method, differing in the basic descriptor, were evaluated on the complete AFF bark dataset:

MS-LBP-KlSVM concatenating histograms of 256 LBPs,
MS-LBP^{u2}-KlSVM concatenating histograms of 59 LBP^{u2},
MS-LBP-HF-KlSVM concatenating 38 LBP-HF features.

Experiment 1 compares the recognition rates using the Fiel-Sablatnig protocol, i.e. with 15 training images per class. This evaluation allows us to compare the results to Fiel, Sablatnig [3] or to the results of Sixta's method [12] which we reimplemented and evaluated on the full AFF bark dataset. Table II shows the recognition results. To prevent results biased by the choice of training data, we performed ten experiments with random selection of training images.

TABLE II. Experiment 1: Recognition accuracy on the AFF bark dataset using the Fiel-Sablatnig protocol

Method	Mean accuracy	σ
MS-LBP-KlSVM	**82.5**%	2.7%
MS-LBP^{u2}-KlSVM	**79.6**%	2.7%
MS-LBP-HF-KlSVM	**74.4**%	3.4%
Sixta [12] (reimplemented)	**51.2**%	2.9%
Fiel, Sablatnig [3]	**64.2**%	–

Experiment 2 is carried out with ten-fold cross validation to provide more training samples - in the set-up, there are about 95 traning images per class on average. Result in TableIII (left) show that the larger training set size leads to a significant, almost 15% increase in recognition accuracy. The confusion matrix for MS-LBP-KlSVM is presented in Table IV.

To simulate the case of image taken under unknown angle, we performed *Experiment 3* with randomly rotated training and testing images. The results in Table III show, as expected, that on the randomly rotated data, the LBP-HF description performs significantly better than LBP and LBP^{u2}.

TABLE III. Experiment 2 and 3: Recognition accuracy on the AFF dataset using ten-fold cross validation

			With random rotation	
Method	Mean accuracy	σ	Mean accuracy	σ
MS-LBP-KlSVM	**96.5**%	2.7%	**84.9**%	4.8%
MS-LBP^{u2}-KlSVM	**94.5**%	2.5%	**82.2**%	3.4%
MS-LBP-HF-KlSVM	**92.2**%	2.7%	**91.0**%	2.1%

The Computer Vision Lab, TU Vienna, kindly made the dataset available to us for academic purposes, with courtesy by Österreichische Bundesforste/Archiv.

The reimplementation was verified on Sixta's incomplete dataset, the results were close to the ones reported in [12].

The rotation was discretized to multiples of 15°.

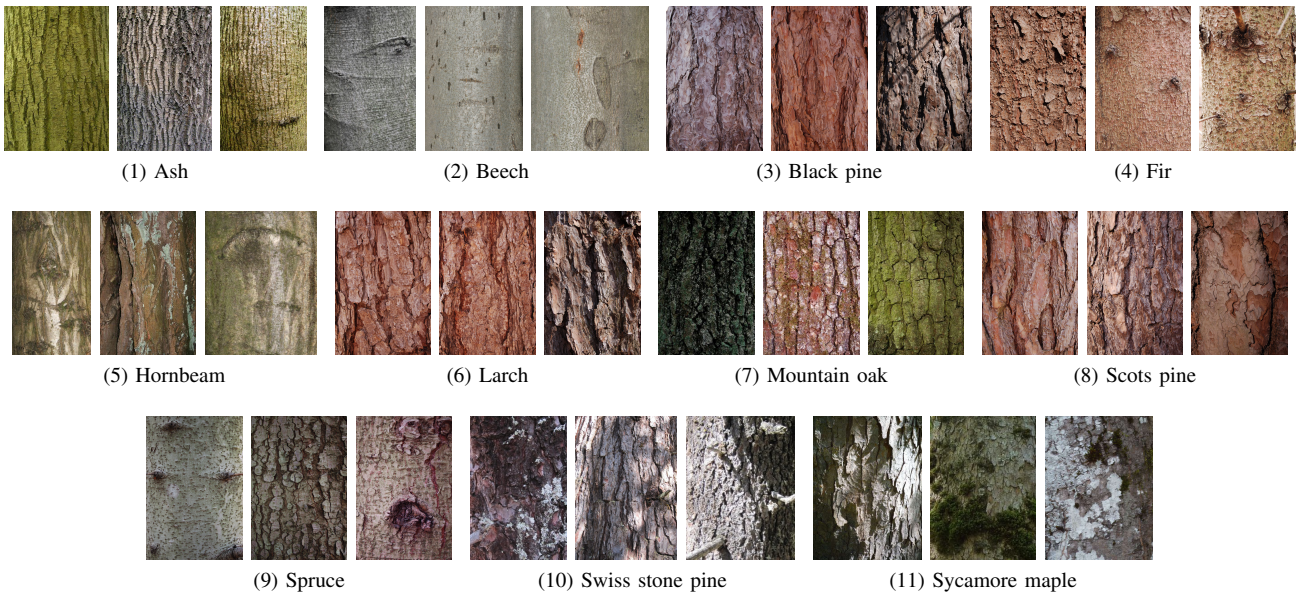


Fig. 4. Examples from the AFF bark dataset

TABLE IV. CONFUSION MATRIX FOR MS-LBP-KLSVM IN EXPERIMENT 2

		Predicted class										
		#1	*#2*	*#3*	*#4*	*#5*	*#6*	*#7*	*#8*	*#9*	*#10*	*#11*
Actual class	*#1*	**33**	0	0	0	0	0	0	0	0	0	0
	#2	0	**13**	0	2	0	0	0	0	1	0	0
	#3	0	0	**162**	0	0	2	0	2	0	0	0
	#4	0	0	0	**123**	0	1	0	1	2	0	0
	#5	0	0	1	0	**40**	1	0	0	0	0	0
	#6	0	0	2	0	0	**194**	0	4	0	0	0
	#7	0	0	0	0	0	0	**75**	0	2	0	0
	#8	0	0	4	0	0	5	0	**181**	0	0	0
	#9	0	0	0	1	0	2	0	0	**210**	0	0
	#10	0	0	0	0	1	1	0	1	1	**91**	1
	#11	0	0	0	0	0	1	0	0	2	2	**17**

TABLE V. RECOGNITION ACCURACY ON THE BRODATZ32 DATASET

Method	**Mean accuracy**	σ
MS-LBP-KlSVM	**97.7**%	0.6%
MS-LBP^{u2}-KlSVM	**96.2**%	0.6%
MS-LBP-HF-KlSVM	**96.2**%	0.6%
Valkealahti et al. [22]	**93.9**%	0.7%
Sharma et al. [25]	**99.5**%	0.2%
LBP-baseline from [25]	**87.3**%	1.5%
RLBP [26]	**98.9**%	–
LBP in [26]	**91.4**%	–
SIFT in [26]	**91.4**%	–
Gabor in [26]	**95.8**%	–

B. Evaluation of the proposed method on standard texture datasets

To check we have not overfitted to the AFF dataset, we evaluated of the proposed method on two standard texture datasets.

Brodatz32 dataset [22] is one of many datasets derived from popular Brodatz textures [23]. It consists of 2048 samples (64×64 images) in 32 classes. The samples are based on 32 Brodatz textures divided into 16 disjoint samples which were additionally rotated by 90° or/and scaled. The experiments were evaluated using one half of the samples for training and the other half for testing, similar to [22], [25], [26]. Table V provides the result comparison.

The second used dataset was the UIUC dataset (UIUCTex) [24] - a dataset consisting of 1000 uncalibrated, unregistered images: 40 samples (640×480) each of 25 different texture classes. The images include significant viewpoint changes, scale differences are present within each class and illumination conditions are uncontrolled. Additional sources of variability present in this dataset include non-planarity of the textured surface, significant non-rigid deformations between different samples of the same class, inhomogeneities of the texture pattern and viewpoint-dependent appearance variations. We provide two standard experiments for the UIUC texture dataset as in [24], [26], [27] - using 10 and 20 training images per class. The result comparison for both evaluation methods is in Table VI.

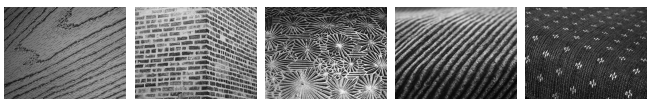


Fig. 6. Examples from the UIUCTex dataset

Fig. 5. Examples from the Brodatz32 dataset

TABLE VI. RECOGNITION ACCURACY ON THE UIUCTEX DATASET

	20 training images per class		10 training images per class	
Method	**Mean accuracy**	σ	**Mean accuracy**	σ
MS-LBP-KlSVM	**91.8**%	1.5%	**80.9**%	3.2%
MS-LBP^{u2}-KlSVM	**89.4**%	1.0%	**78.6**%	2.9%
MS-LBP-HF-KlSVM	**96.4**%	0.6%	**92.6**%	1.0%
Lazebnik [24]	**96.0**%	–	**92.6**%	–
Zhang [27]	**99.0**%	–	–	
RLBP [26]	–		**96.7**%	–
LBP in [26]	–		**92.7**%	–
SIFT in [26]	–		**92.8**%	–
Gabor in [26]	–		**93.8**%	–

IV. CONCLUSIONS

We proposed a method for bark texture recognition based on multi-scale texture description using concatenated LBP histograms. Robustness to scale changes is handled by computing multiple descriptors on different scales. Linear SVM classifier was chosen for its fast evaluation and low memory footprint. We have shown that using a feature map approximating the histogram intersection kernel significantly improves the methods accuracy.

Experimental results show that the proposed method significantly outperforms the state-of-the-art [3]. On the AFF bark dataset, when training with ten-fold cross validation, the method reached 96.5% accuracy. Using 15 images per class for training, the proposed method achieved 82.5% accuracy compared with 64.2% reported in [3] and the 51.2% for reimplementation of [12]. The proposed algorithm has better results than both human experts [3], whose accuracy was 56.6% and 77.8%. All performed experiments have also shown, that the common practice of using only uniform LBPs causes a decrease in accuracy.

Experiments on standard texture datasets (Brodatz32, UIUC) have shown the advantages of rotation invariant LBP-HF description and confirmed that the proposed method is suitable for general texture recognition and that it performs close to the state-of-the-art.

ACKNOWLEDGMENT

JM was supported by the Czech Science Foundation Project GACR P103/12/G084, MS by CTU Project SGS13/142/OHK3/2T/13 and Technology Agency of the Czech Republic Project TE01020415 V3C.

REFERENCES

[1] P. N. Belhumeur, D. Chen, S. Feiner, D. W. Jacobs, W. J. Kress, H. Ling, I. Lopez, R. Ramamoorthi, S. Sheorey, S. White, and L. Zhang, "Searching the world's herbaria: A system for visual identification of plant species," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 116–129.

[2] M.-E. Nilsback and A. Zisserman, "An automatic visual flora: Segmentation and classification of flower images," Ph.D. dissertation, Oxford University, 2009.

[3] S. Fiel and R. Sablatnig, "Automated identification of tree species from images of the bark, leaves and needles," in *Proc. of 16th Computer Vision Winter Workshop*, Mitterberg, Austria, 2011, pp. 1–6.

[4] N. Kumar, P. N. Belhumeur, A. Biswas, D. W. Jacobs, W. J. Kress, I. C. Lopez, and J. V. Soares, "Leafsnap: A computer vision system for automatic plant species identification," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 502–516.

[5] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in large margin classifiers*, vol. 10, no. 3, 1999.

[6] H.-T. Lin, C.-J. Lin, and R. C. Weng, "A note on platt's probabilistic outputs for support vector machines," *Machine learning*, vol. 68, no. 3, 2007.

[7] A. Vedaldi and A. Zisserman, "Efficient additive kernels via explicit feature maps," *PAMI*, vol. 34, no. 3, 2011.

[8] Z. Chi, L. Houqiang, and W. Chao, "Plant species recognition based on bark patterns using novel gabor filter banks," in *ICNNSP, in Proc.*, vol. 2, 2003.

[9] Y.-Y. Wan, J.-X. Du, D.-S. Huang, Z. Chi, Y.-M. Cheung, X.-F. Wang, and G.-J. Zhang, "Bark texture feature extraction based on statistical texture analysis," in *ISIMP, in Proc.*, 2004.

[10] J. Song, Z. Chi, J. Liu, and H. Fu, "Bark classification by combining grayscale and binary texture features," in *ISIMP, in Proc.*, 2004.

[11] Z.-K. Huang, C.-H. Zheng, J.-X. Du, and Y.-y. Wan, "Bark classification based on textural features using artificial neural networks," in *Advances in Neural Networks-ISNN'2006*. Springer, 2006.

[12] T. Sixta, "Image and video-based recognition of natural objects," Diploma Thesis, Czech Technical University in Prague, 2011.

[13] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in *IAPR 1994, in Proc.*, vol. 1, 1994, pp. 582–585 vol.1.

[14] T. Ojala, M. Pietikäinen, 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.

[15] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *PAMI*, vol. 24, no. 7, pp. 971–987, 2002.

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

[17] T. Mäenpää and M. Pietikäinen, "Multi-scale binary patterns for texture analysis," in *Image Analysis*. Springer, 2003, pp. 885–892.

[18] M. Pietikäinen, T. Ojala, and Z. Xu, "Rotation-invariant texture classification using feature distributions," *Pattern Recognition*, vol. 33, no. 1, pp. 43–52, 2000.

[19] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, "Rotation invariant image description with local binary pattern histogram fourier features," in *SCIA '09, in Proc.* Springer-Verlag, 2009, pp. 61–70.

[20] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

[21] V. Franc and S. Sonnenburg, "Optimized cutting plane algorithm for support vector machines," in *ICML'08*. ACM, 2008, pp. 320–327.

[22] K. Valkealahti and E. Oja, "Reduced multidimensional co-occurrence histograms in texture classification," *PAMI*, vol. 20, no. 1, pp. 90–94, 1998.

[23] P. Brodatz, *Textures: a photographic album for artists and designers*. Dover New York, 1966, vol. 66.

[24] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine regions," *PAMI*, vol. 27, no. 8, pp. 1265–1278, 2005.

[25] G. Sharma, S. ul Hussain, and F. Jurie, "Local higher-order statistics (lhs) for texture categorization and facial analysis," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 1–12.

[26] J. Chen, V. Kellokumpu, G. Zhao, and M. Pietikäinen, "Rlbp: Robust local binary pattern," in *BMVC*, 2013.

[27] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *IJCV*, vol. 73, no. 2, pp. 213–238, 2007.