Feature Maps for Approximation of Additive Kernels in Support Vector Machines.

Milan Šulc

Center for Machine Perception
FEE CTU in Prague

PhD supervisor: Jiří Matas

8.10.2015

Visual Classification of Textures, Applications in Plant Recognition.

- Texture: common feature of natural objects

Plant Recognition

- Natural object recognition
  - non-rigid structures, high intra-class variance

- Original motivation: <u>Intelligent field guide as a mobile app</u>

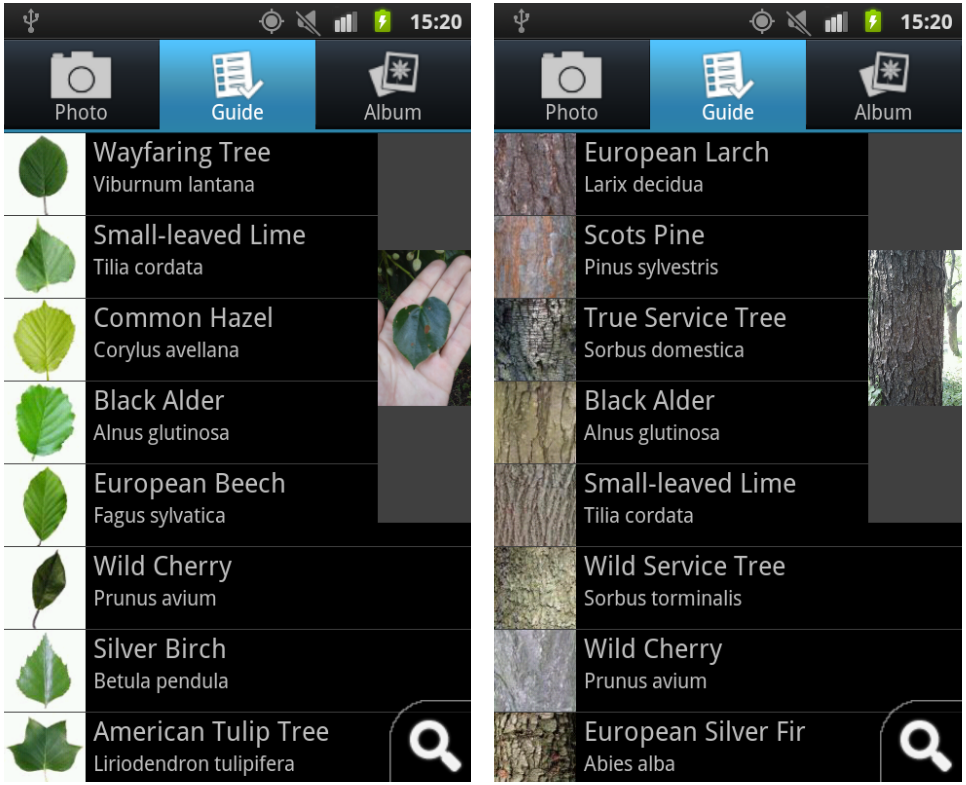


Fig.1: developed Android application

Practical requirements:

- Fast (ideally realtime)
- Precise for higher number of classes (>100)
- Low storage load (reasonable app size)

Bark Identification: Textural Problem

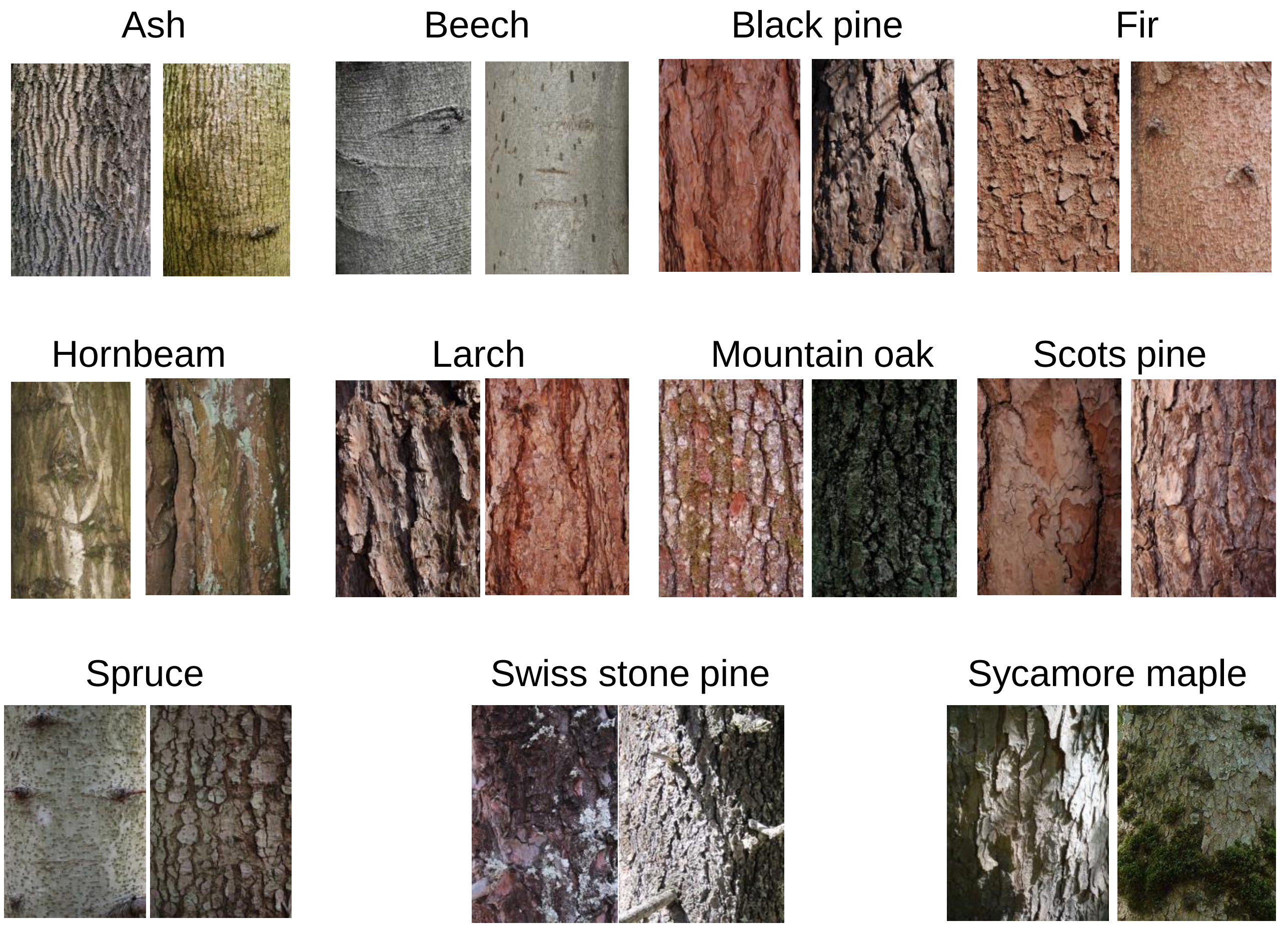


Fig.2: Examples from the Austrian Federal Forests dataset [1]

[1] Automated identification of tree species from images of the bark, leaves and needles.
S. Fiel and R. Sablatnig, in Proc. of 16th CVWW

Leaf Identification: Standard Approaches

Fig.3: Common problems with leaf description

- Methods for leaf recognition commonly describe:
  - Shape
  - Color
  - Local feature points (e.g. SIFT)
  - <u>Texture</u>

The Ffirst Method [2]

The **Ffirst** **(Fast Features Invariant to Rotation and Scale of Texture)** method for texture classification uses several state-of-the-art approaches:

1) Fast description: histograms of (Completed) Local Binary Patterns

2) Rotation-invariant representation: “histogram Fourier features”

3) Improved scale space for multi-scale description and scale invariance

4) Linear SVM classifiers with feature maps approx. the intersection kernel

[2] Fast Features Invariant to Rotation and Scale of Texture.
M. Sulc and J. Matas, ECCV 2014 Workshops (LBP'14)

Local Binary Patterns (LBP)

- Compares the intensity of each pixel to its neighborhood

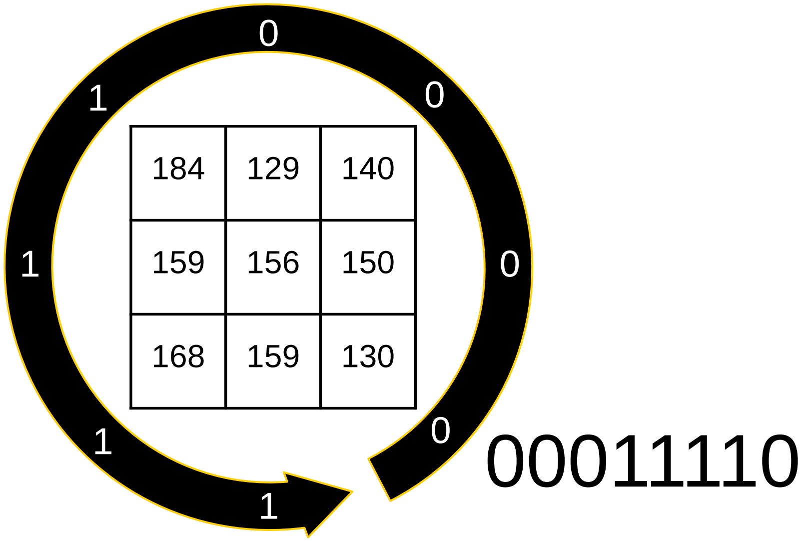


Fig.4: LBP operator

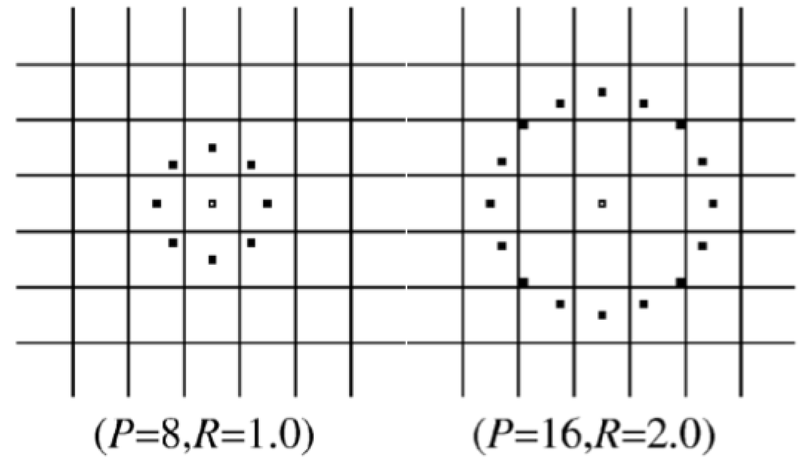


Fig. 5: Examples of common neighborhoods

- 58 **uniform** patterns from 256 $\mathrm{LBP}_{8,R}$

$$\mathrm{LBP}_{P,R}(x,y) = \sum_{p=0}^{P-1} s(f(x,y) - f(x_p,y_p))2^p, \quad s(x) = \begin{cases} 1: & x \leq 0 \\ 0: & \text{else} \end{cases}$$

Completed Local Binary Patterns (CLBP)

1) **LBP-S** = Standard LBP (Sign-LBP)

2) **LBP-M** = Magnitude-LBP introduced:
binary thresholding the intensity difference magnitudes

$$\text{LBP-M}_{P,R}(x,y) = \sum_{p=0}^{P-1} s(|f(x,y) - f(x_p,y_p)| - t_p)2^p$$

$$t_p = \sum_{i=1}^{m} \frac{|f(x_i,y_i) - f(x_{ip},y_{ip})|}{m}$$

[3] A completed modeling of local binary pattern operator for texture classification.
Guo, Z., Zhang, D. Image Processing, IEEE Transactions on 19(6), 1657–1663 (2010)

Rotation Invariance on Uniform LBP

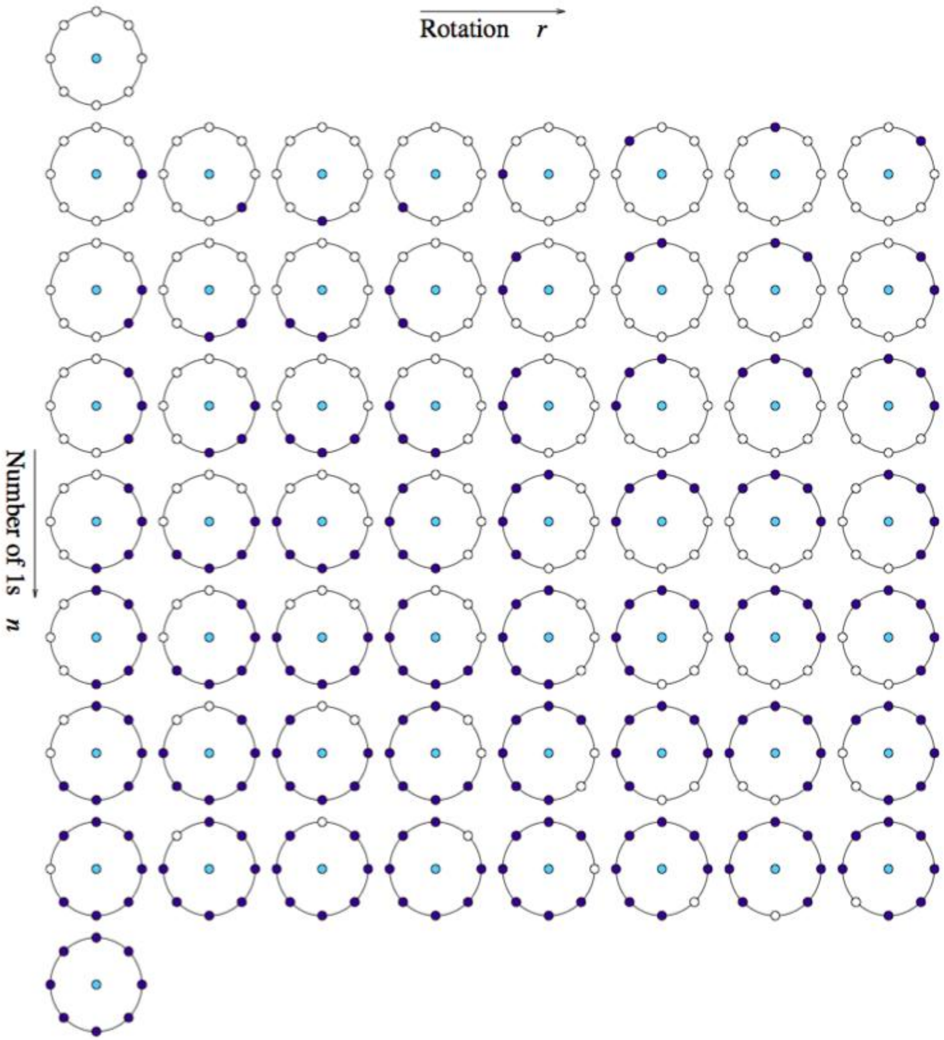
- Standard LBP^{riu2}:
  - Drops the rotation r

- LBP-HF (Histogram Fourier features)
  - Performs FFT for each row

$$H(n,u) = \sum_{r=0}^{P-1} h_I(U_p^{n,r}) e^{-i2\pi ur/P}$$

  - FFT magnitudes are rotation-invariant

$$|H(n,u)| = \sqrt{H(n,u)\overline{H(n,u)}}$$



[4] Rotation invariant image description with local binary pattern histogram fourier features.
Ahonen, T., Matas, J., He, C., Pietikainen, M. SCIA '09, in Proc. (2009)

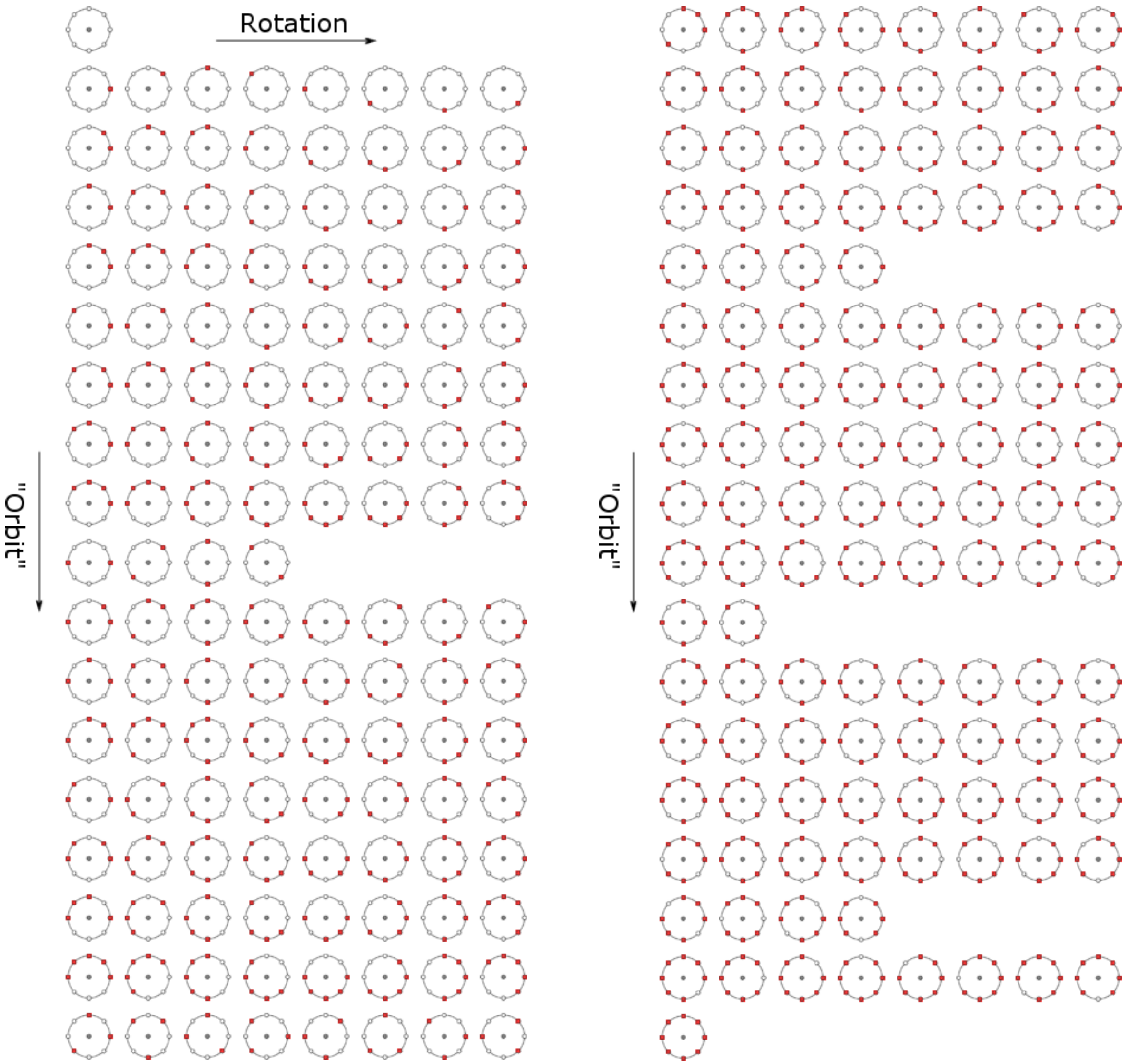
Extending the LBP-HF Features

1) Use the full set of LBP to compute LBP-HF features

2) Additional LBP-HF+ features

- Built from the first harmonics of 2 orbits

$$\text{LBP-HF}^{+}(n) = \sqrt{H(n,1)\overline{H(n+1,1)}}$$

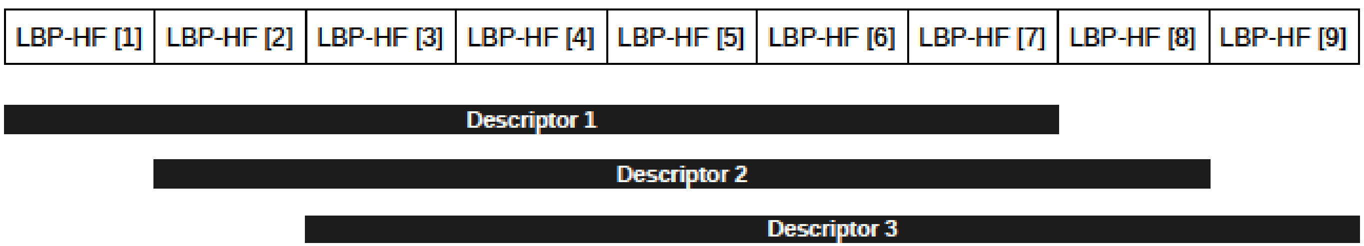


[2] Fast Features Invariant to Rotation and Scale of Texture.
M. Sulc and J. Matas, ECCV 2014 Workshops (LBP'14)

Scale Space

1) Features from different scales concatenated into a **multi-scale descriptor**.

2) The multi-scale descriptor is computed over different scale ranges.



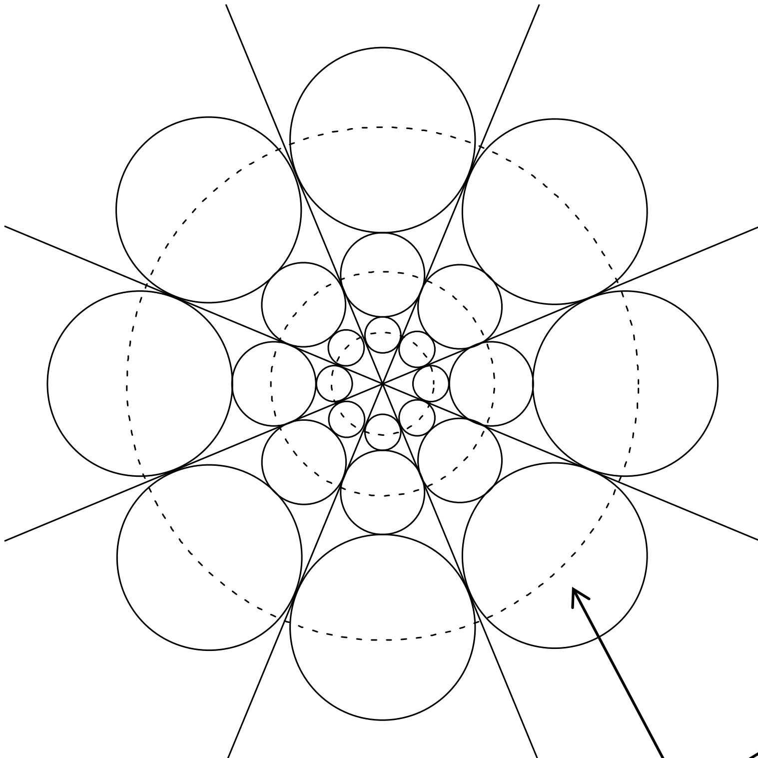
⇨ 9 LBP scales

⇨ 3 descriptors

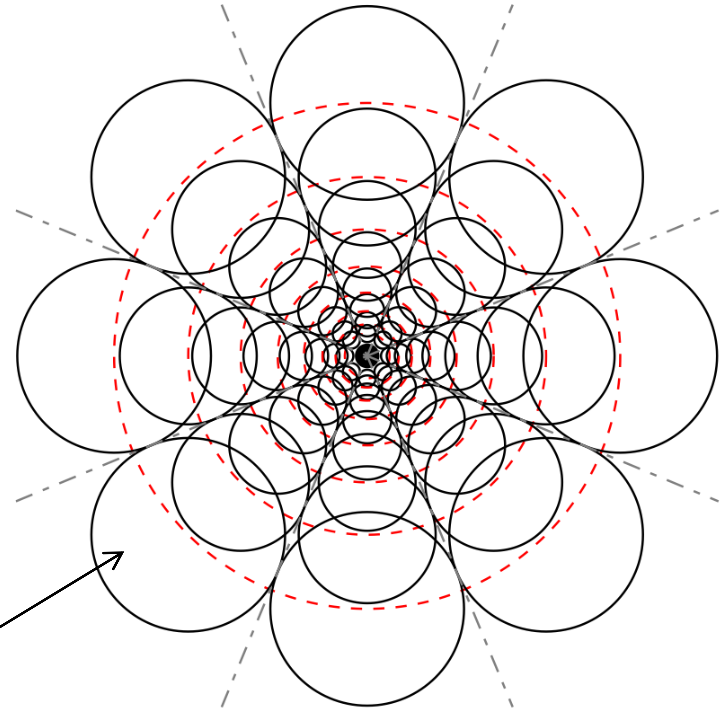
exponential radius growth VS. sufficient number of scales

Scale Space

Standard LBP [5] scale space:



Our scale space [2]:



Gaussian blur
(effective areas)

$$R_i = R_{i-1}\sqrt{2} \qquad r_i = R_i \sin\frac{p}{P}$$

[2] Fast Features Invariant to Rotation and Scale of Texture.
M. Sulc and J. Matas, ECCV 2014 Workshops (LBP'14)
[5] Multi-scale binary patterns for texture analysis.
T. Maenpaa and M. Pietikainen, in *Image Analysis*. Springer, 2003, pp. 885–892.

Learning and Classification

- Linear SVMs learned on feature-mapped data (approximating the intersection kernel) [6]

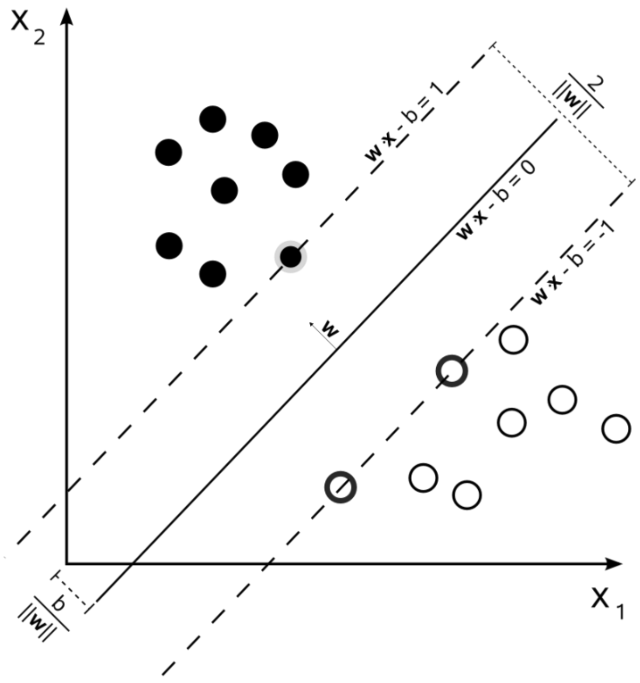
- Combined using the "One-vs-All" scheme

- Platt's probabilistic output [7], numerically stable version [8]
  - ➢posterior probability estimate

- Result: class (and scale) with highest posterior probability estimate

[6] Efficient Additive Kernels via Explicit Feature Maps
A. Vedaldi and A. Zisserman, *PAMI*, vol. 34, no. 3, 2011.
[7] Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods.
J. Platt, *Advances in large margin classifiers*, vol. 10, no. 3, 1999.
[8] A note on platt's probabilistic outputs for support vector machines.
H.-T. Lin, C.-J. Lin, and R. C. Weng, *Machine learning*, vol. 68, no. 3, 2007.

Support Vector Machines



Separating hyperplane, evaluation:

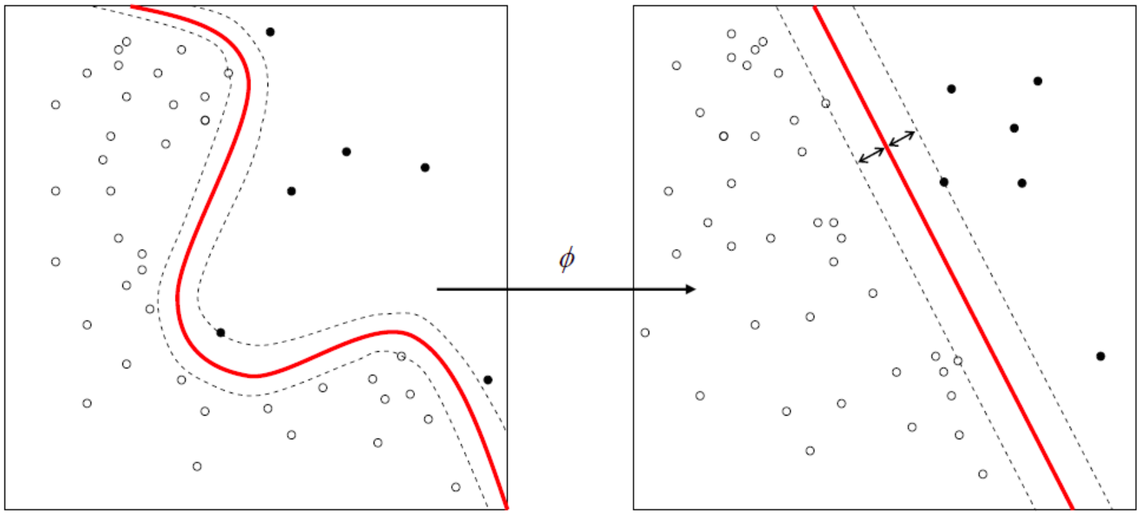
$$\mathrm{F}(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle$$

Learned by minimizing $E(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{n}\sum_{i=1}^{n} \ell_i(\langle \mathbf{w}, \mathbf{x} \rangle)$ (primal)

or maximizing $D(\boldsymbol{\alpha}) = -\frac{1}{2\lambda n^2}\boldsymbol{\alpha}^\top X^\top X \boldsymbol{\alpha} + \frac{1}{n}\sum_{i=1}^{n} -\ell_i^*(-\alpha_i)$ (dual)

$$\mathbf{w}(\boldsymbol{\alpha}) = \frac{1}{\lambda n}\sum_{i=1}^{n} \mathbf{x}_i \alpha_i = \frac{1}{\lambda n} X \boldsymbol{\alpha}$$

The Kernel Trick



$$F(\boldsymbol{x}) = \sum_i \beta_i K(\boldsymbol{x}, \boldsymbol{x}_i)$$

Non-linear SVM: replace inner product $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ by a kernel function $K(\boldsymbol{a}, \boldsymbol{b})$

There exists a feature map $\psi(\boldsymbol{x})$ mapping the data to a Hilbert space $\mathcal{H}$

such that $K(a, b) = \langle \psi(\boldsymbol{a}), \psi(\boldsymbol{b}) \rangle_{\mathcal{H}}$

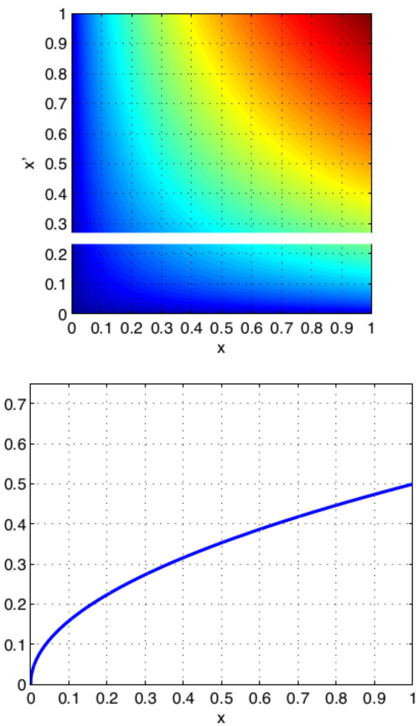
Additive Homogeneous Kernels

Additive kernel
Sum of 1D kernels

$$K(\mathbf{x}, \mathbf{x}') = \sum_{l=1}^{D} k(x_l, x'_l)$$

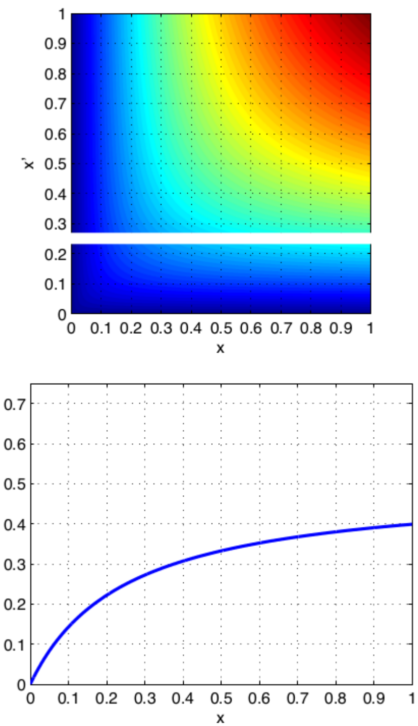
Hellinger

$$k(x, x') = \sqrt{xx'}$$



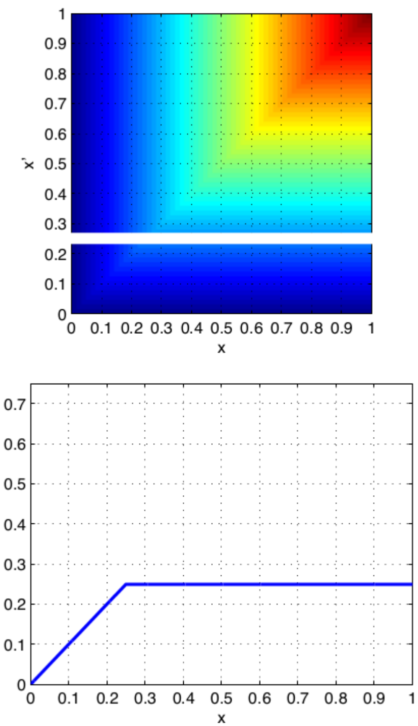
X²

$$\frac{2xx'}{x + x'}$$



intersection

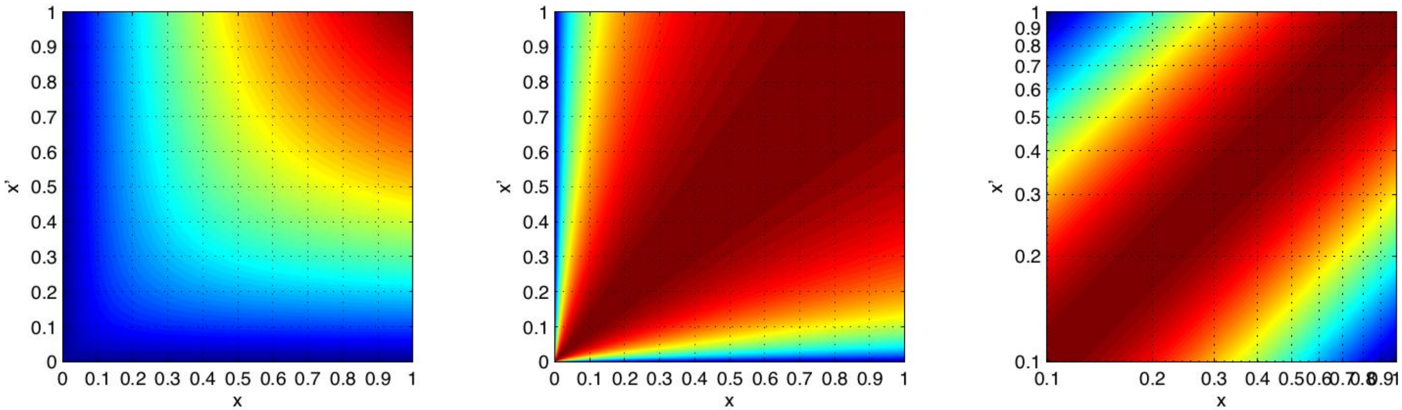
$$\min\{x, x'\}$$



(Slide from CVPR 2013 tutorial given by A. Vedaldi)

Additive Homogeneous Kernels: Trick

$k(x, x')$ | $\dfrac{k(x, x')}{\sqrt{xx'}}$ | $\log x$



Homogeneous kernel
Multiplicative constant pops out

$$\forall c \geq 0 : k(cx, cx') = ck(x, x')$$

Signature / profile
Up to a factor and a logarithm

$$k(x, x') = \sqrt{xx'}\, \mathcal{K}(\log x - \log x')$$

$$\boxed{\Phi_\omega(x) = \kappa_\omega \sqrt{x}\, e^{-\mathbf{i}\langle \omega, \log x \rangle}}$$

(Slide from CVPR 2013 tutorial given by A. Vedaldi)

Additive Homogeneous Kernels: Examples

	Hellinger	X²	intersection
	$k(x,x') = \sqrt{xx'}$	$\dfrac{2xx'}{x+x'}$	$\min\{x,x'\}$
	$\mathcal{K}(\lambda) = 1$	$e^{-\lvert\lambda\rvert/2}$	$\operatorname{sech}(\lambda/2)$
	$\kappa_\omega^2 = \delta(\omega)$	$\dfrac{2}{\pi(1+4\omega^2)}$	$\operatorname{sech}(\pi\omega)$
	$\Phi_\omega(x) = \sqrt{x}$	$\sqrt{\dfrac{2x}{\pi(1+4\omega^2)}}e^{-\mathbf{i}\,\omega\log x}$	$\sqrt{x\operatorname{sech}(\pi\omega)}e^{-\mathbf{i}\,\omega\log x}$

(Slide from CVPR 2013 tutorial given by A. Vedaldi)

Approximate Feature Maps

- Consider a periodicization of the kernel signature

$$\hat{\mathcal{K}}(\lambda) = \operatorname*{per}_{\Lambda} W(\lambda)\mathcal{K}(\lambda) = \sum_{k=\infty}^{+\infty} W(\lambda + k\Lambda)\mathcal{K}(\lambda + k\Lambda)$$

- This gives a discrete version of Bochner's result:

$$\hat{\mathcal{K}}(\lambda) = \sum_{j=-\infty}^{+\infty} \hat{\kappa}_j e^{-ijL\lambda}$$

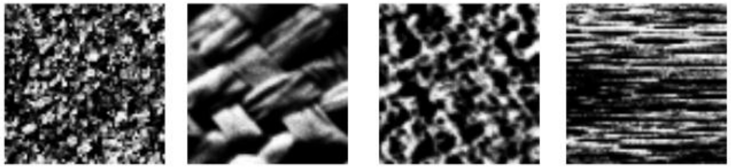
- Then, for homogeneous kernels:

$$\boxed{\hat{\Psi}_j(x) = \begin{cases} \sqrt{x^\gamma \hat{\kappa}_0}, & j = 0, \\ \sqrt{2x^\gamma \hat{\kappa}_{\frac{j+1}{2}}} \cos\left(\frac{j+1}{2} L \log x\right) & j > 0 \text{ odd}, \\ \sqrt{2x^\gamma \hat{\kappa}_{\frac{j}{2}}} \sin\left(\frac{j}{2} L \log x\right) & j > 0 \text{ even}, \end{cases}}$$

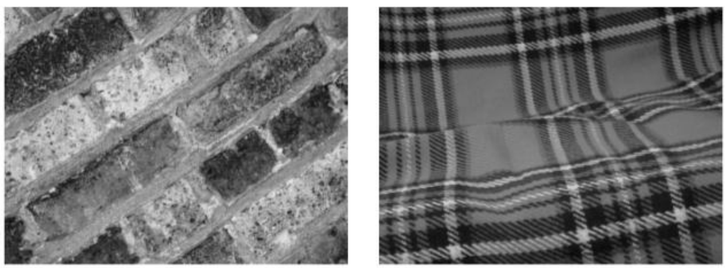
[6] Efficient Additive Kernels via Explicit Feature Maps
A. Vedaldi and A. Zisserman, *PAMI*, vol. 34, no. 3, 2011.

Texture Classification

- Brodatz32



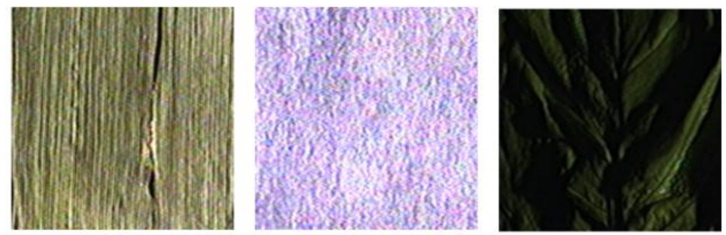
- UIUCTex



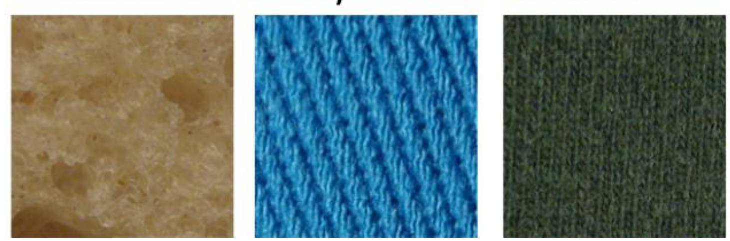
- UMD

- ALOT

- CUReT



- KTH-TIPS
- KTH-TIPS2a, KTH-TIPS2b



Results: Texture Classification

	Brodatz32	*UIUCTex*	*UMD*	*CUReT*	*ALOT*	*KTH-TIPS*	*KTH-TIPS2a*	*KTH-TIPS2b*
# classes	32	25	25	61	250	10	11	11
Ffirst∀+	99.4±0.3	99.4±0.4	99.3±0.3	**99.7±0.1**	96.4±0.2	99.5±0.5	**87.9±6.1**	76.6±4.3
FV-VGG-VD [9]	-	**99.9±0.1**	**99.9±0.1**	99.0±0.2	**98.5±0.1**	**99.8±0.2**	-	**81.8±2.5**
IFV_{SIFT} [10]	-	97.0±0.9	99.2±0.4	99.6±0.3	-	99.7±0.1	82.5±5.2	69.3±1.0
Best results	**99.5±0.2** [11]	-	-	-	-	-	-	-

Table 1: Texture classification accuracy

[9] Deep filter banks for texture recognition, description, and segmentation.
Cimpoi, M., Maji, S., Kokkinos, I., Vedaldi, A. arXiv preprint arXiv:1507.02620 (2015).

[10] Describing textures in the wild.
Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A. Computer Vision and Pattern Recognition (2013)

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

Results: Texture Classification Speed

Ffirst∀+	**0.048 s / im.**
IFV_{SIFT} [9]	0.466 s / im.
FV-VGG-VD [8]	4.910 s / im.

Table 2: Average description time (200x200px images)

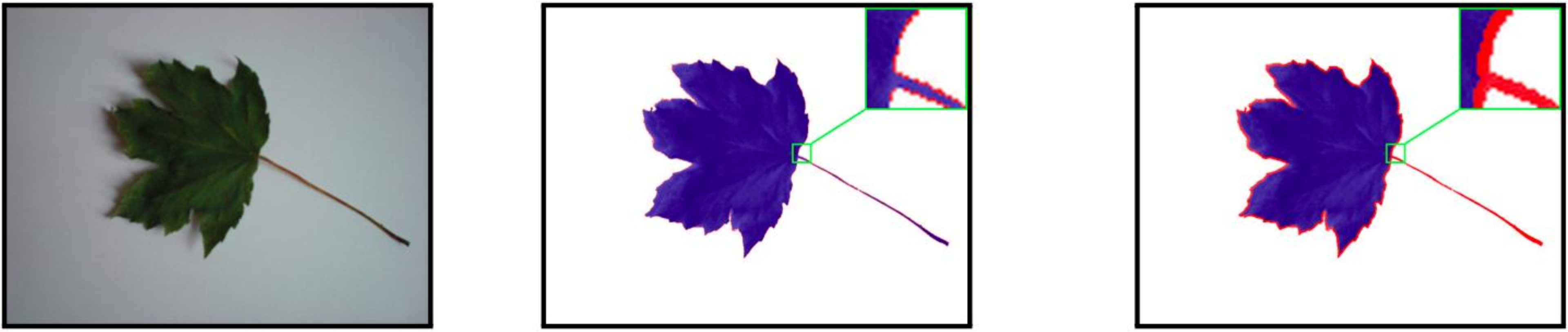
- MATLAB implementations using the VLFeat and MatCovNet library
- without parallelization / GPU !
- implementation of [8,9] kindly provided by the authors

[9] Deep filter banks for texture recognition, description, and segmentation.
Cimpoi, M., Maji, S., Kokkinos, I., Vedaldi, A. arXiv preprint arXiv:1507.02620 (2015).

[10] Describing textures in the wild.
Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A. Computer Vision and Pattern Recognition (2013)

Describing Leaves

- All datasets contain leaves on a white background
- Simple segmentation by thresholding + filling holes



- Define leaf border as the area, where at least one neighbor in LBP doesn’t belong to the segmented foreground
- Idea: describe the leaf interior and border separately

Results: Leaf Classification

	AFF	Flavia 10 : 40	Flavia ½ : ½	Foliage	Swedish	MEW	Leafsnap	Leafsnap (top5)
# of classes	5	32	32	60	15	153	185	185
Ffirst$\forall+^{i}$	97.3±1.5	99.3±0.3	98.9±0.3	98.1	99.6±0.4	98.4±0.2	73.1±2.3	92.4±1.7
Ffirst$\forall+^{b}$	99.5±0.6	99.3±0.4	99.0±0.2	98.3	99.4±0.5	97.9±0.2	77.2±1.9	94.8±1.5
Ffirst$\forall+^{ib}_{\prod}$	**100.0±0.0**	**99.8±0.3**	**99.7±0.1**	**99.3**	**99.8±0.3**	**99.5±0.1**	**83.7±1.1**	**97.3±1.1**
Best results	93.6 [1]	97.2 [12]	96.5 [13]	95.8 [14]	99.4 [15]	84.9 [16]	≈ 73 [17]	96.8 [17]

Table 3: Leaf classification accuracy

[1] Automated identification of tree species from images of the bark, leaves and needles.
S. Fiel and R. Sablatnig, in Proc. of 16th CVWW

[12] An implementation of leaf recognition system.
Lee, K.B., Chung, K.W., Hong, K.S. (2013)

[13] An efficient representation of shape for object recognition and classification using circular shift method
Karuna, G., Sujatha, B., GIET, R., Reddy, P.C. IJSER (2013)

[14] Performance improvement of leaf identification system using principal component analysis.
Kadir, A., Nugroho, L.E., Susanto, A., Santosa, P.I. IJAST (2012)

[15] Pairwise rotation invariant co-occurrence local binary pattern.
Qi, X., Xiao, R., Guo, J., Zhang, L. ECCV 2012, pp. 158–171. Springer (2012)

[16] Leaf recognition of woody species in central europe.
Novotný, P., Suk, T. Biosystems Engineering 115(4), 444–452 (2013)

[17] Leafsnap: A computer vision system for automatic plant species identification.
Kumar, N, Belhumeur, P. N., Biswas, A., Jacobs, D. W., Kress, W. J., Lopez, I. C. Soares, J. V. ECCV (2012)

Results: Plant Phenotyping

- **Using a subset of images from the LifeCLEF'14 plant ident. task**
  - Species, observation ID, GPS information
  - We use one image per observations from NORTH and SOUTH of France
  - Our task: given the class, recognize region (NORTH / SOUTH)

	Betula pendula Roth	Corylus avellana L.	Castanea sativa Mill.	Acer campestre L.
Ffirst^{i}	85.0 **%**	**95.0 %**	85.0 **%**	70.0 **%**
Ffirst^{b}	**90.0 %**	80.0 **%**	80.0 **%**	75.0 **%**
$\mathrm{Ffirst}^{ib}_{\sqcap}$	**90.0 %**	85.0 **%**	**90.0 %**	**85.0 %**

Table 4: Leaf-based tree location classification, 10-fold cross validation

Results: Bark Classification

Austrian Federal Forests dataset: 11 classes, 1182 images in total

Method	*Accuracy* [%]
Ffirst∀+	84.9 ± 2.5
SIFT, BoW [1]	64.2
AFF forest ranger [1]	77.8 *
AFF botanist [1]	56.6 *

* *human experts were tested on a smaller image set*

Table 5: Bark classification accuracy using 15 training images per class (Fiel-Sablatnig protocol)

Method	*Accuracy* [%]
Ffirst∀+	96.5 ± 1.2

Table 6: Bark classification accuracy using 10-fold cross validation

[1] Automated identification of tree species from images of the bark, leaves and needles. S. Fiel and R. Sablatnig, in Proc. of 16th CVWW

Summary of the Results

Fast Features Invariant to Rotation and Scale of Texture

- Texture classification based on LBP obtains excellent results
  - >=99% accuracy on most datasets
  - cca 100x faster than the state-of-the-art method

Plant identification

- Leaf classification using textural features (Ffirst)
  - >= 99% accuracy on most datasets
  - significantly outperforms the best reported results

- Bark classification
  - dataset deficiency
  - best reported results on the Austrian Federal Forests dataset, outperforming also the accuracy of both human experts

- Thank you **!** Questions **?**

sulcmila@fel.cvut.cz