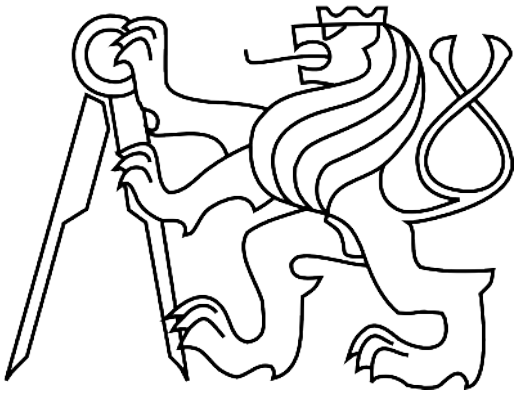
Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics



DIPLOMA THESIS

Tree Identification from Images

Milan Šulc

Supervisor: Prof. Ing. Jiří Matas, Ph.D.

Study Programme: Open Informatics

May, 2014

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

DIPLOMA THESIS ASSIGNMENT

Student: Bc. Milan Š u l c

Study programme: Open Informatics

Specialisation: Computer Vision and Image Processing

Title of Diploma Thesis: Tree Identification from Images

Guidelines:

1. Familiarize yourself with the state-of-the-art in image-based identification of trees.
2. Build a system for visual identification of bark, leaves and their combination.
3. The system should be applicable to mobile devices. Semi-automatic methods may be used, utilizing interactive user input.
4. Implement the methods and evaluate them on publicly available data.

Bibliography/Sources:

[1] P. N. Belhumeur et al.: Searching the world's herbaria: A system for visual identification of plant species, ECCV 2008.
[2] S. Fiel and R. Sablatnig: Automated identification of tree species from images of the bark, leaves and needles, CVWW 2011.
[3] T. Sixta: Image and Video-based Recognition of Natural Objects, Diploma thesis, 2011.
[4] N. Kumar et al., Leafsnap: A computer vision system for automatic plant species identification, ECCV 2012.

Diploma Thesis Supervisor: prof. Ing. Jiří Matas, Ph.D.

Valid until: the end of the summer semester of academic year 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 10, 2014

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Bc. Milan Š u l c

Studijní program: Otevřená informatika (magisterský)

Obor: Počítačové vidění a digitální obraz

Název tématu: Identifikace stromů z obrázků

Pokyny pro vypracování:

1. Seznamte se s moderními postupy identifikace stromů z obrazu.
2. Sestavte systém pro vizuální identifikaci podle kůry, listů a jejich kombinací. Tento systém by měl být aplikovatelný na mobilních zařízeních. Mohou být použity poloautomatické metody využívající interaktivní vstup od uživatele.
3. Metody implementujte a vyhodnoťte na veřejně dostupných datech.

Seznam odborné literatury:

[1] P. N. Belhumeur et al.: Searching the world's herbaria: A system for visual identification of plant species, ECCV 2008.
[2] S. Fiel and R. Sablatnig: Automated identification of tree species from images of the bark, leaves and needles, CVWW 2011.
[3] T. Sixta: Image and Video-based Recognition of Natural Objects, Diploma thesis, 2011.
[4] N. Kumar et al.: Leafsnap: A computer vision system for automatic plant species identification, ECCV 2012.

Vedoucí diplomové práce: prof. Ing. Jiří Matas, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 10. 1. 2014

Acknowledgement

I would like to express the greatest appreciation to my supervisor Jiří Matas for his patient guidance throughout this thesis and the amount of time he spent on consultations. His suggestions, ideas and comments were invaluable.

I would also like to thank other people providing valuable assistance and support, including my colleagues from the Center for Machine Perception and my friends.

Finally, I am deeply grateful to my family for their endless support and encouragement.

Prohlášení autora práce

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Dne ....

Podpis autora práce

Abstract

This thesis focuses on the problem of automatic identification of tree species based on images of leaves and bark. We propose to describe both leaves and bark using textural features. FSRIT (Fast Scale- and Rotation- Invariant Texture), a novel method for texture description and recognition, is introduced. The method combines an improved scale space (used for multi-scale representation and scale invariant matching) with several state-of-the-art approaches (including LBP-HF features and use of linear SVM classifiers with approximate kernel map). Using the proposed method we achieve new state of the art results in the classification of bark (Austrian Federal Forests bark dataset) and leaves (Austrian Federal Forests leaf dataset, Flavia dataset, Foliage dataset, Swedish dataset and Middle European Woods dataset), as well as on standard textural datasets KTH-TIPS2a and KTH-TIPS2b, while achieving 99% accuracy on all other standard textural datasets (KTH-TIPS, CUReT, UIUCTex, UMD and Brodatz32). The proposed recognition method is very fast and thus suitable for real time applications, including e.g. mobile field guides for plant identification.

Anotace

Tato práce je zaměřena na problém automatického rozpoznávání druhu stromu dle obrázků listů a kůry. Navrhujeme popisovat kůru i listy pomocí texturálních příznaků. Představujeme FSRIT (Fast Scale- and Rotation- Invariant Texture), novou metodu pro popis a rozpoznávání textury. Ta kombinuje vylepšený prostor měřítek (pro multi-scale reprezentaci a scale-invariantní párování) s několika moderními přístupy (včetně LBP-HF příznaků či použití lineárních SVM klasifikátorů s aproximatiními kernelovými mapami). S použitím této navržené metody dosahujeme nejlepších výsledků v klasifikaci kůry (na datasetu kůry Austrian Federal Forests) a listů (na datasetech listů Austrian Federal Forests, Flavia, Foliage, Swedish a Middle European Woods), jakož i na standardních texturálních datasetech KTH-TIPS2a a KTH-TIPS2b, dosahujíce 99% přesnosti na všech ostatních standardních texturálních datasetech (KTH-TIPS, CUReT, UIUCTex, UMD and Brodatz32). Navržená metoda je přitom velmi rychlá a tedy vhodná pro aplikace v reálném čase, včetně např. mobilního klíče pro rozpoznávání rostlin.

Contents

1 Introduction **10**

1.1 The proposed approach to leaf and bark recognition 11

1.2 Structure of the thesis 12

2 State of the art **14**

2.1 Plant recognition 14

2.1.1 Bark recognition 14

2.1.2 Leaf recognition 15

2.1.3 Combination of leaves and bark 18

2.1.4 Other images of plants 19

2.2 Texture recognition 20

3 The Fast Scale and Rotation Invariant Texture (FSRIT) recognition method **21**

3.1 Local Binary Patterns 21

3.2 Rotation invariant description 21

3.3 Additional information using Magnitude-LBP 23

3.4 Multi-scale description and scale invatiance 24

3.5 Support Vector Machine and feature maps 26

4 Experiments **27**

4.1 Texture classification 28

4.1.1 Brodatz32 28

4.1.2 UIUCTex 29

4.1.3 KTH-TIPS 29

4.1.4 KTH-TIPS2 30

4.1.5 CUReT 30

4.1.6 UMD 31

4.2 Suitability for real-time applications 32

4.3 Recognition of leaves and bark 33

4.3.1 Austrian Federal Forest (AFF) datasets 33

4.3.2 Flavia leaf dataset 35

4.3.3 Foliage leaf dataset 36

4.3.4 Swedish leaf dataset 37

4.3.5 Middle European Woods (MEW) dataset 37
4.4 Combining leaf and bark 39
4.5 Species retrieval and classification errors 40

5 Conclusions **42**
5.1 Future work 43

A Contents of the enclosed DVD **52**

B Publicly available datasets **52**

C Visualisation of misclassified samples **53**

List of Figures

1 Examples of possible leaf description problems, illustrated on images from Flavia [1] and Swedish dataset [2] 12
2 The Leafsnap application for iOS devices 17
3 Illustration of the $\mathrm{LBP}_{P,R}$ operator neighbourhoods from [3] 21
4 Illustration of the uniform $\mathrm{LBP}_{8,1}$ patterns ordered by their n,r, as used to compute the histogram Fourier features from [4] 22
5 Rotation invariant LBP-HF features computed from the uniform LBP, illustrated on tree bark 23
6 The effective areas of filtered pixel samples in a multi-resolution $\mathrm{LBP}_{8,R}$ operator 25
7 Feature mapping and concatenating features from multiple scales in FSRIT, KTH-TIPS2b 27
8 Dependency on the number c of multiscale descriptors in FSRIT, KTH-TIPS2b 28
9 Examples of 4 texture classes from the UIUCTex database 29
10 Examples of 4 texture classes from the KTH-TIPS2 database 30
11 Examples of 4 texture classes from the CUReT database 31
12 Examples of 4 texture classes from the UMD database 32
13 Examples from the AFF bark dataset 34
14 Examples from the AFF leaf dataset 35
15 Examples of 4 classes from the Flavia leaf dataset 36
16 Examples of 4 classes from the Foliage dataset 36

17 Examples of 4 classes from the Swedish dataset 37
18 Examples of 4 classes from the MEW dataset 38
19 Retrieval precision for different lengths of shortlist, MEW leaf dataset (153 classes) 40
20 Retrieval precision for different lengths of shortlist, AFF bark dataset (11 species) 41
21 Examples of misclassification on the AFF bark dataset 53
22 Examples of misclassification on the MEW leaf dataset 54

List of Tables

1 Evaluation of FSRIT on the KTH-TIPS datasets, compared to the state-of-the-art methods 31
2 Evaluation of FSRIT on other standard datasets, compared to the state-of-the-art methods 32
3 Average image description time on selected datasets, compared to IFV_{SIFT} 33
4 Evaluation of FSRIT on the AFF datasets of leaves and bark using the Fiel-Sablatnig protocol 34
5 Evaluation of FSRIT on the AFF datasets of leaves and bark using 10 fold cross validation 35
6 Evaluation of FSRIT on the leaf datasets (Flavia, Foliage, Swedish, Middle European Woods) 38
7 Evaluation of FSRIT on the AFF datasets, combining both leaves and bark classification 40
8 DVD contents description 52
9 Leaf datasets 52
10 Texture datasets 52

List of Algorithms

1 The FSRIT description method overview 26

1 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 [5–8]. The application potential of plant recognition has increased as devices equipped with cameras became ubiquitous, making intelligent field guides, education tools and automation in forestry and agriculture practical.

The so called intelligent field guides, using computer vision algorithms to make plant identification easier, can be very helpful to botanists in the field, as discussed by Belhumeur et al. [5]:

> *"Botanists in the field are racing to capture the complexity of the Earth's flora before climate change and development erase their living record. To greatly speed up the process of plant species identification, collection, and monitoring, botanists need to have the world's herbaria at their fingertips."*

Such systems, sorting thousands of species to show the relevant results, can significantly speed up the plant identification process, which in some cases would take hours or days using a standard dichotomous key, even for specialists.

One of the key steps needed to create a system supporting the efforts of biologists and botanists in the study of biodiversity on Earth is the development of fast and precise image recognition methods for plant species classification, or rather plant species retrieval.

The resulting intelligent field guides using such methods can also be used in education and by non-expert users, being simple to use and providing more user comfort than traditional field guides (usually based on dichotomous keys for plant identification).

Although we also mention other possible applications of plant image analysis, including automatic fruit inspection and quality evaluation or automatic identification of disease symptoms, the focus of our work is on the problems of recognizing tree bark and leaves from images and using the combination of bark and leaves for tree classification. We develop a new automatic plant identification method taking into account the requirements for practical applications. It is worth noting that, even though we primarily define and evaluate the problems as image classification, the proposed recognition method is actually very suitable for the retrieval too, using the "one vs. all" classification scheme for Support Vector Machines with probabilistic output.

1.1 The proposed approach to leaf and bark recognition

We propose to approach both the bark and leaf recognition as a texture recognition problem.

While using only textural features for description is a common approach for bark recognition, as discussed in Section 2.1.1, most leaf recognition methods are based on features of different nature, often several of them combined together. However, as discussed in Section 2.1.2, some of the previous methods also rely on texture description only.

To describe an image of leaf, we first need to perform a binary segmentation on the input image, separating the leaf from the background. In practical applications it is often too complicated or even impossible to take the picture on a light background and the result is then strongly dependent on the quality of leaf segmentation, especially in the case of shape features, where even details as leaf serration (illustrated in Figure 1a) play an important role. One of the advantages of a recognition system based on textural description is that the segmentation precision is not as critical in comparison to shape description.

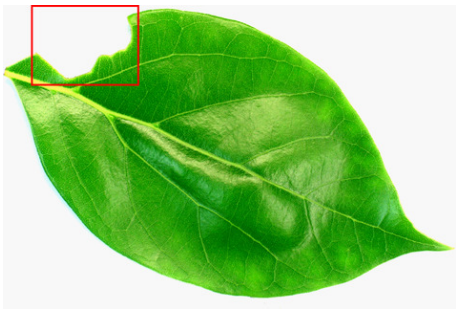
Shape description methods also have problems with recognition of damaged leaves (Figure 1c) and with correct segmentation and description of compound leaves (Figure 1b). The leaf texture, on the other hand, may be damaged by a disease, as in Figure 1d.

To build a recognition system which is fast, accurate and robust to scale change, we introduce a novel texture description and recognition system combining several state-of-the-art methods with a novel scale space setting for Local Binary Patterns. The proposed Fast Scale- and Rotation-Invariant Texture (FSRIT) recognition method is described step by step in Section 3.

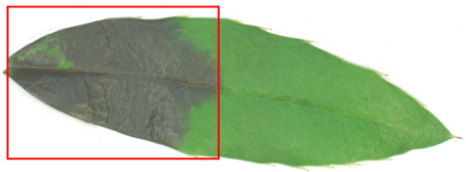
On pictures of bark, the whole image area is described using FSRIT, whereas in the case of leaves only the leaf area is described. The leaf datasets used in our experiments in Section 4.3 contain leaves scanned or photographed on a light (or white) background. Our experimental setting can thus use a simple thresholding method, namely the Otsu's method [9]. To deal with complicated background in practical applications, any fast state-of-the-art segmentation methods can be utilized, in the case of mobile applications the segmentation method can also ask for user input or correction (as proposed by Sixta [10]).

(a) Leaf serration segmentation

(b) Compound leaf



(c) Damaged leaf shape



(d) Disease

Figure 1: Examples of possible leaf description problems, illustrated on images from Flavia [1] and Swedish dataset [2]

1.2 Structure of the thesis

The thesis is structured as follows:

Section 2.1 provides a short overview of the computer vision approaches to plant recognition. We propose to describe both bark and leaves using textural features. Practical advantages of texture description for leaves, which is not a widely used approach, were discussed in Section 1.1. A quick introduction to texture identification and the state-of-the-art methods used for texture recognition problems is given in Section 2.2. A novel texture recognition method satisfying the requirements for application in bark and leaf classification is introduced in Section 3. This Fast Scale and Rotation Invariant Texture (FSRIT) recognition method is compared to the state-of-the-art texture recognition meth-

ods in Section 4.1. The suitability of FSRIT to bark and leaf recognition is experimentally verified in Section 4.3, mostly in terms of plant classification accuracy, but we also provide more detail results in terms of plant retrieval, i.e. the probabilities of retrieving the correct species within the first N results. The retrieval experiments are performed in Section 4.5 on both bark images, using the AFF bark dataset, and leaf images, using MEW (the largest leaf dataset in our experiments).

Classification using the combination of leaf and bark images is discussed and the improvement in accuracy achieved by this combination is shown in experiments in Section 4.4. All plant classification experiments are also repeated using a rotation dependent version of the method (denoted as FSIT), which could achieve better results given input images acquired in a standard orientation (i.e. "up is up" orientation for tree bark images).

The results are discussed and conclusions are made in Section 5.

2 State of the art

2.1 Plant recognition

2.1.1 Bark recognition

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

Chi et al. [11] proposed a method using Gabor filter banks. Their accuracy reaches 96% on a dataset with 8 classes, containing 25 images (132×132 px) per class. The dataset was not published. Wan et al. [12] performed a comparative study of bark texture features. Four approaches were tested: the grey 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 authors also show that the performance of all classifiers improved significantly, up to 89% using COMM and 1-NN, when colour information was added.

Song et al. [13] 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. [14] 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 is not available, it is difficult to assess the quality of the results and to perform comparative evaluation.

Fiel and Sablatnig [7] proposed methods for automated identification of tree species from images of the bark, leaves and needles. For bark description they created a Bag of Words with SIFT descriptors in combination with GCLM and wavelet features. A Support Vector Machine with radial basis function (RBF) kernel was used for classification. They 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 and will be used in our experiments in Sections 4.3.1 and 4.4. A recognition accuracy of 64.2% and 69.7% was achieved on this dataset for training sets with 15 and

30 images per class (where available).

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 his diploma thesis [10] 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, which was provided to him by the authors of [7]. With two-fold cross-validation, the recognition rate on this dataset was 70.1%.

One of the issues of the bark recognition problem is the public dataset deficiency. Luckily, the full version of the AFF dataset was kindly provided to us by the Computer Vision Lab, TU Vienna, for academic purposes, with courtesy by Österreichische Bundesforste/Archiv.

We published a preliminary version of the proposed method for bark recognition, not including the magnitude-LBP and using different solver settings with histogram intersection kernel map instead of the χ^2 kernel map, as a conference paper [15] in 2013.

2.1.2 Leaf recognition

Recognition of leaves usually denotes only recognition of broad leaves, whereas needles are treated separately. Several techniques have been proposed for leaf description, often based on combining features of different character (shape features, colour features, etc.).

As mentioned above, Sixta [10] also proposed a method for leaf recognition, using the IDSC (Inner Distance Shape Context) for shape recognition. To find the leaf orientation for efficient shape matching, Centroid Contour Distance (CCD) curve has been utilised. A 83.3% accuracy was measured on the Flavia database randomly divided into two halves (training set and testing set).

The leaf recognition method by Fiel and Sablatnig [7] is based on a Bag of Words model with SIFT descriptors and achieves 93.6% accuracy on the AFF leaf dataset. The

dataset contains 134 broad leaf images of 5 Austrian tree species, with 25 to 34 images per species, and has also been kindly provided to us, even though it is not publicly available. We use this datasets for our experiments in Sections 4.3.1 and 4.4

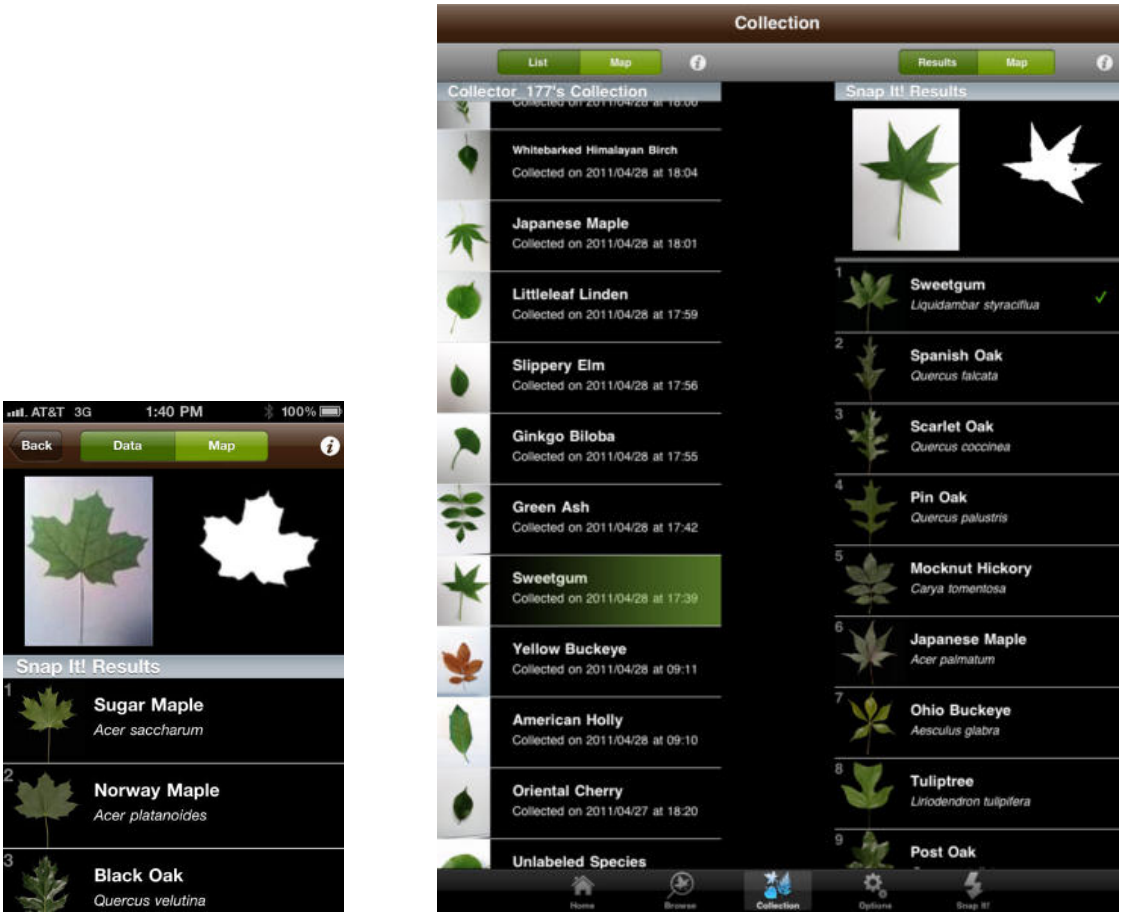
Kadir et al. compare several shape methods in recognizing plants [16]. From the compared methods (geometric features, moment invariants, Zernike moments, Polar Fourier Transform) Polar Fourier Transform performed best achieving 64% in accuracy on a database of 52 plant species, each represented by 20 leaf samples. Unfortunately the dataset was not published.

Kumar et al. [8] describe Leafsnap, a computer vision system for automatic plant species identification, which has evolved from the previously proposed plant identification systems by Agarwal et al. [17] and Belhumeur et al. [5]. Compared to the previous versions, they introduced a pre-filter on input images, numerous speed-ups and additional post-processing within the segmentation algorithm, the use of a simpler and more efficient curvature-based recognition algorithm instead of Inner Distance Shape Context (IDSC); a larger dataset of images, and a new interactive system for use by non-expert users. They introduce the Leafsnap database of 184 tree species, which is presented as public, however at the time of writing this thesis it was not available online[1]. On this database, 96.8% of queries have a species match within the top 5 results shown to the user with the used method. The resulting electronic field guide, developed by researches from Columbia University, the University of Maryland, and the Smithsonian Institution, is available as a free mobile app for iOS devices. Although the app (screenshots in Figure 2) runs on iPhone and iPad devices, the leaf images are actually processed at a server, thus internet connection is required for recognition, which might cause problems in natural areas with slow or no data connection. Another requirement is to take the photos of the leaves on a fully-white background.

Wu et al. [1] proposed a Probabilistic Neural Network for leaf recognition using 12 commonly used digital morphological features (DMFs), derived from 5 basic features (diameter, physiological length, physiological width, leaf area, leaf perimeter). The authors also collected a publicly available database of plant leaves called Flavia, containing 1907 images of leaves from 32 species. The average accuracy on the current version of the dataset is 93% (according to the Flavia web page[2]). The Flavia dataset is discussed in Section 4.3.2, together with the comparison of our results to the best reported results by

[1]The author's responded they have not been able to prepare the database in a suitable form.

[2]`http://flavia.sourceforge.net`



(a) The iPhone version (b) The iPad version makes use of the larger screen area

Figure 2: The Leafsnap application for iOS devices

Kadir et al. [18, 19] and Lee et al. [20, 21], as well as to the results reported by Sixta [10], Novotný and Suk [22], and Karuna et al. [23], who used a different evaluation method.

Kadir et al. [24] prepared the Foliage dataset, consisting of 60 classes of leaves, each of them containing 120 images. The comparison of results on the Foliage dataset is available in Section 4.3.3. The best reported result on this dataset by Kadir et al. [18] was achieved by a combination of shape, vein, texture and colour features processed by Principal Component Analysis before classification by a Probabilistic Neural Network.

Söderkvist [2] proposed a computer vision system classification of leaves from Swedish trees and collected the so called Swedish dataset. The dataset contains scanned images of 15 tree classes, 75 samples each. Wu et al. [25] introduced a visual descriptor for scene categorization called spatial Principal component Analysis of Census Transform (spatial PACT), achieving a 97.9% accuracy on the Swedish dataset. Qi et al. (according to their

website[3]) achieve 99.38% accuracy on the Swedish dataset using their texture descriptor called PRI-CoLBP [26] with SVM classification. In Section 4.3.4 we provide experimental results on the Swedish dataset.

Novotný and Suk [22] proposed a leaf recognition system for identification of Central European woody species, using Fourier descriptors of the leaf contour normalised to translation, rotation, scaling and starting point of the boundary. The authors also collected a new large leaf dataset called Middle European Woods (MEW) containing 153 classes (from 151 botanical species) of native or frequently cultivated trees and shrubs of the Central Europe Region. The dataset contains at least 50 samples per species and a total of 9745 samples. When dividing this dataset into two halves (training and testing set), their method achieves 84.92% accuracy.

One of the possible applications of leaf description is the identification of a disease symptom. Pydipati et al. [27] proposed a system for identification of citrus disease using CCM textural features, achieving accuracies of over 95% for 4 classes (normal leaf samples and diseased citrus leaf samples with greasy spot, melanose, and scab).

2.1.3 Combination of leaves and bark

Kim et al. [28] proposed tree classification using a combination of leaf, flower and bark photos of the same tree. They used 20 features of wavelet decomposition with 3 levels for a grey and a binary image for description of bark, 32 features of Fourier descriptor for leaf and 72 features of HS colour plane for flower. The results were tested on their own (unpublished) dataset consisting of 16 classes. They report 31%, 75% and 75% recognition accuracy for individual leaf, flower and bark classification and 84%, 75% and 100% accuracy for combinations of leaf+flower, leaf+bark and bark+flower. However, in all cases only one image per class was tested (without any repetition using other images), which makes the results unreliable and misleading.

Although experiments with both leaves and bark have been performed by Fiel and Sablatnig [7], they did not carry out experiments with the combination of leaf and bark images, relying on sufficiently accurate results achieved using the leaf images only.

Pl@ntNet[4] is an interactive plant identification and collaborative information system providing an image sharing and retrieval application for plant identification. It is developed by scientists from four French research organisations (Cirad, INRA, Inria and IRD) and

[3] `http://qixianbiao.github.io`

[4] `http://www.plantnet-project.org/`

the Tela Botanica network. The Pl@ntNet-identify Tree Database provides identification by combining information from images of the habitat, flower, fruit, leaf and bark, but the exact algorithms used in the Pl@ntNet-identify web service[5] and their accuracies are not publicly documented.

2.1.4 Other images of plants

Several other approaches have been proposed to recognize trees and plants from images of their organs.

Nilsback and Zisserman [6] investigated the segmentation and classification of flower images. They introduced 2 new datasets for flower classification, the smaller of them containing 17 categories and the larger containing 102 categories. A recognition rate of 75.3% was achieved on the larger dataset, where in 95.2% cases the correct class was among first 10 results. Flower recognition was also studied by Kim et al. [29], who proposed a mobile based flower recognition system and implemented it for Windows Mobile 5.0 Pocket PC. Other flower description and recognition methods were described by Saitoh et al. [30, 30], Pornpanomchai et al. [31, 32] and Cho et Lim [33]. Mattos et al. [34] recently described an efficient approach for flower classification that is suitable for deployment in mobile devices.

Fiel and Sablatnig [7] also discussed a method for recognition of needles (or more precisely coniferous branches), and introduced the AFF needle dataset, which included 275 images of the needles of the 6 most common Austrian conifers. They concluded that no method has been found for the classification of the needle images and proposed a method to distinguish at least between fir and spruce. A semi-automatic method for coniferous branches recognition and a dataset of coniferous branches from 4 common Czech conifers was published in my bachelor thesis [35], but the classification results were not very reliable, achieving 72.3% accuracy among 4 classes.

Arivazhagan et al. [36] proposed a fusion of colour and texture features for fruit recognition. Yang et al. [37] presented a fruit recognition method for automatic harvesting.

Brosnan and Sun [38] provided a review on the possibilities of computer vision in food inspection, including the inspection and grading of fruits and vegetables. Kim et al. [39] investigated the potential of using colour texture features for detecting citrus peel diseases, achieving best results (96.7% classification accuracy among 6 classes of grapefruits - with

[5] `http://identify.plantnet-project.org/en/`

normal and five common diseased peel conditions) using 14 selected HSI texture features. Another systems for visual identification and classification of affected and normal fruits was presented by Pujari et al. [40]. The developments in the application of artificial vision systems for the automatic inspection and quality evaluation of fruits and vegetables were presented by Cubero et al. [41].

2.2 Texture recognition

Texture description and recognition techniques have been the subject of many studies for their wide range of applications. The early work focused on the problem of terrain analysis [42,43] and material inspection [44]. Later applications of texture analysis include face recognition [45], facial expressions [46, 47] and object recognition [48]. The relation between scene identification and texture recognition is discussed by Renninger and Malik [49]. Texture analysis is a standard problem with several surveys available, e.g. [50–52].

Several recent approaches report fine results on the standard datasets, often using complex texture description methods. A cascade of invariants computed using scattering transforms is used to construct an affine invariant texture representation by Sifre and Mallat [53]. A sparse representation based Earth Mover's Distance (SR-EMD) presented by Li et al. [54] achieves good results in both image retrieval and texture recognition. Local Higher-Order Statistics (LHS) proposed by Sharma et al. [47] describe higher-order differential statistics of local non-binarized pixel patterns. The method by Cimpoi et al. [55] uses Improved Fisher Vectors (IFV) for texture description. This work also shows further improvement when combined with describable texture attributes learned on the Describable Textures Dataset (DTD).

Many texture description methods are based on the Local Binary Patterns [3, 56–62], which is a computationally simple and powerful approach.

3 The Fast Scale and Rotation Invariant Texture (FSRIT) recognition method

3.1 Local Binary Patterns

The FSRIT description is based on the well known Local Binary Patterns [56, 57]. The common LBP operator (further denoted as sign-LBP) computes the signs of differences between the 3×3 neighbourhood and the center pixel. The operator can be simply extended [3] to work with arbitrary number of neighbours P on a circle of radius R (Figure 3). The computation of the $\mathrm{LBP}_{P,R}$ for a center point (x, y) is expressed in Eq. 1, using an image function $f(x, y)$ and neighbourhood point coordinates (x_p, y_p).

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

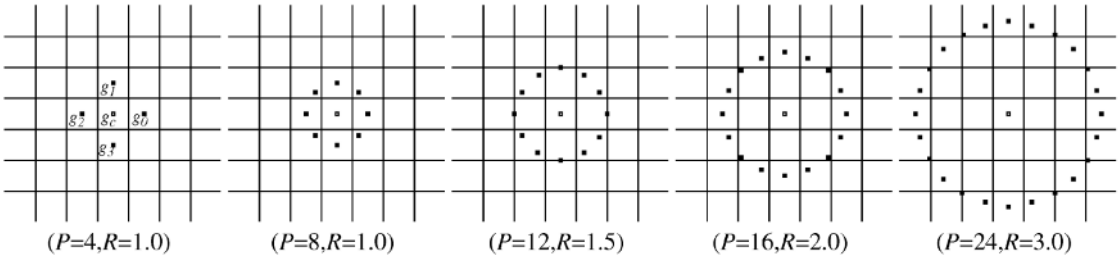


Figure 3: Illustration of the $\mathrm{LBP}_{P,R}$ operator neighbourhoods from [3]

The $\mathrm{LBP}_{8,1}$ operator is similar to the standard LBP operator on a 3×3 neighbourhood, only the values of the diagonal pixels are determined by interpolation and the pixels in the neighbour set are indexed so that they form a circular chain.

3.2 Rotation invariant description

To achieve rotation invariance[6], we use the so called LBP Histogram Fourier Features (LBP-HF) introduced by Ahonen et al. [4]. The Histogram Fourier Features are a highly discriminative rotation invariant description for the histogram of uniform patterns built using the discrete Fourier transform (DFT). Uniform LBP are patterns with at most 2 spatial transitions (bitwise 0-1 changes), as illustrated in Figure 4. Unlike the simple rotation invariant description using $\mathrm{LBP}^{\mathrm{ri}}$ [3, 63], which assigns all uniform patterns with

[6] LBP-HF (as well as LBP^{ri}) are rotation invariant only in the sense of a circular bit-wise shift, e.g. rotation by multiples 22.5° for $\mathrm{LBP}_{16,R}$.

the same number of 1s into one bin (Equation 2), the LBP-HF features preserve the information about relative rotation of the patterns.

$$\mathrm{LBP}^{ri}_{P,R} = \min\left\{\mathrm{ROR}\left(\mathrm{LBP}_{P,R}, i\right) \mid i = 0, 1, .., P-1\right\} \tag{2}$$

Denoting a uniform pattern $U_p^{n,r}$, where n is the number of true bits and r denotes the rotation of the pattern, the DFT for given n can be expressed as:

$$H(n,u) = \sum_{r=0}^{P-1} h_I(U_p^{n,r}) e^{-i2\pi ur/P} \tag{3}$$

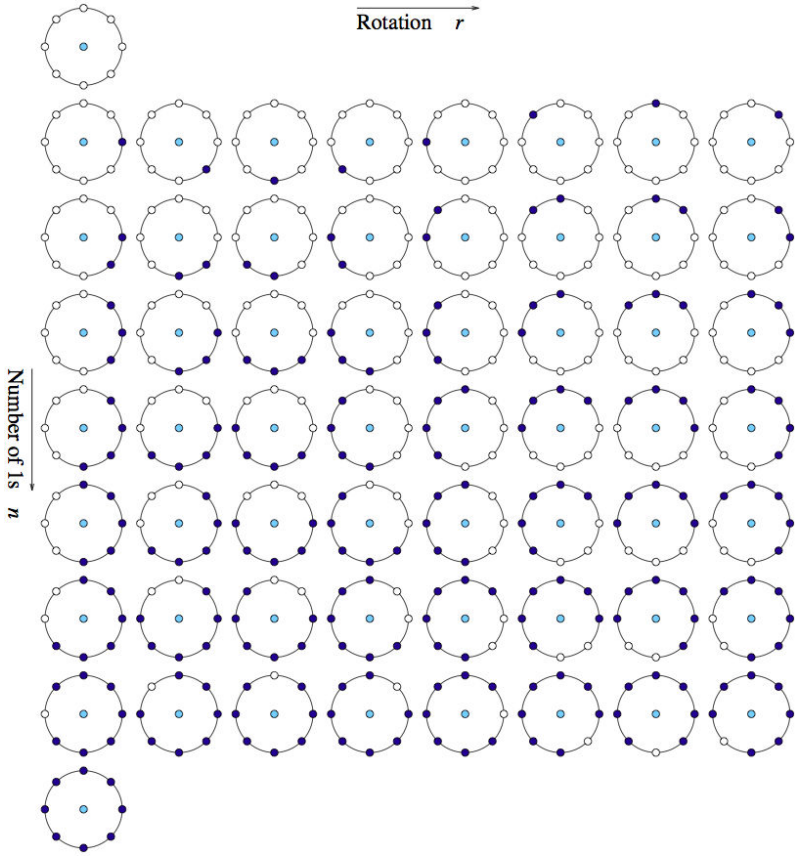


Figure 4: Illustration of the uniform $\mathrm{LBP}_{8,1}$ patterns ordered by their n,r, as used to compute the histogram Fourier features from [4]

where the histogram value $h_I(U_p^{n,r})$ denotes the number of occurrences of given uniform pattern in the image.

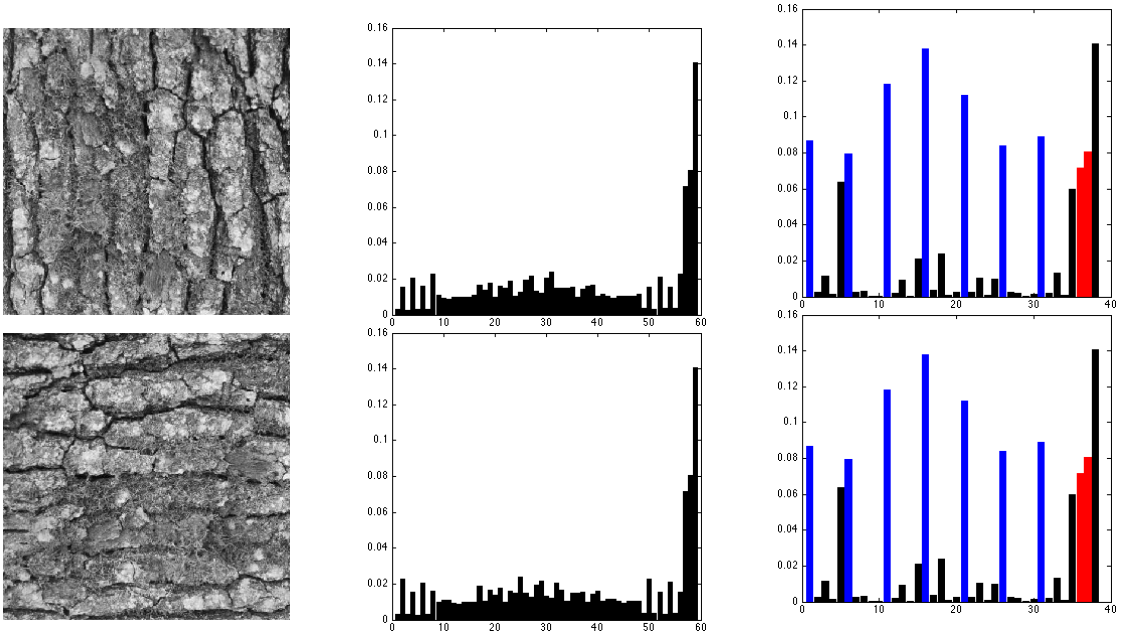
The Fourier magnitude spectrum (respectively LBP-HF features) is then computed as the absolute value of the DFT magnitudes, which corresponds to multiplying the DFT

magnitudes by their complex conjugate values (preventing the phase shift introduced by rotation):

$$|H(n,u)| = \sqrt{H(n,u)\overline{H(n,u)}} \tag{4}$$

Because the result is symmetrical, only one half (more precisely $\lfloor \frac{P}{2} \rfloor + 1$) of the DFT magnitudes are used for each set of uniform patterns with n true bits. Three other bins are then added to the resulting histogram, namely two for the "1-uniform" patterns (with all bins of the same value) and one for all non-uniform patterns.

It is also worth noting that the LBP-HF features contain all LBP^{ri} features - two of them representing the 1-uniform patterns (red bins in Figure 5c) and the rest (blue bins in Figure 5c) being contained in the DFT magnitudes.



(a) Input image (b) Histogram of uniform $\text{LBP}_{8,1}$ features (c) Histogram of LBP-HF features

Figure 5: Rotation invariant LBP-HF features computed from the uniform LBP, illustrated on tree bark

3.3 Additional information using Magnitude-LBP

The LBP histogram Fourier features can be generalized for any set of uniform features. In FSRIT the LBP-HF-S-M description introduced by Zhao et al. [60] is used, where

the histogram Fourier features of both sign- and magnitude-LBP are used to build the descriptor. The combination of both sign- and magnitude-LBP was introduced by Guo and Zhang [62] and denoted as Completed Local Binary Patterns (CLBP). The so called magnitude-LBP describes whether the magnitude of the difference of the neighbouring pixel (x_p, y_p) against the central pixel (x, y) exceeds a threshold t_p:

$$\text{LBP-M}_{P,R}(x, y) = \sum_{p=0}^{P-1} s(|f(x, y) - f(x_p, y_p)| - t_p)2^p \tag{5}$$

We use the common practice of choosing the threshold value (for neighbours at p-th bit) as the mean value of all m differences in the whole image:

$$t_p = \sum_{i=1}^{m} \frac{|f(x_i, y_i) - f(x_{ip}, y_{ip})|}{m} \tag{6}$$

The LBP-HF-S-M histogram is created by simply joining histograms of LBP-HF-S and LBP-HF-M (computed from uniform sign-LBP and magnitude-LBP).

In Section 4 we also experiment with a variant of the method not requiring the rotation invariance. This variant of the method is using the original CLBP histogram (i.e. concatenated sign-LBP and magnitude-LBP histograms) instead of LBP-HF-S-M and is denoted as FSIT (Fast Scale-Invariant Texture).

3.4 Multi-scale description and scale invatiance

A scale space is built by computing LBP-HF-S-M from circular neighbourhoods with exponentially growing radius R. Gaussian filtering is used[7] to overcome pixel noise and gather intensity information from the pixels surroundings.

Unlike the MS-LBP approach by Mäenpää and Pietikäinen [59], where the radii of the LBP operators are chosen so that the effective areas of different scales touch each other, as in Figure 6a, we use a finer scaling with a $\sqrt{2}$ step between scales radii R_i:

$$R_i = R_{i-1}\sqrt{2} \tag{7}$$

This radius change illustrated in Figure 6b is equivalent to decreasing the image area to one half. A finer scale change like that allows us to capture more information about the texture. The first LBP radius used is $R_1 = 1$, as the LBP with low radii capture important high frequency texture characteristics.

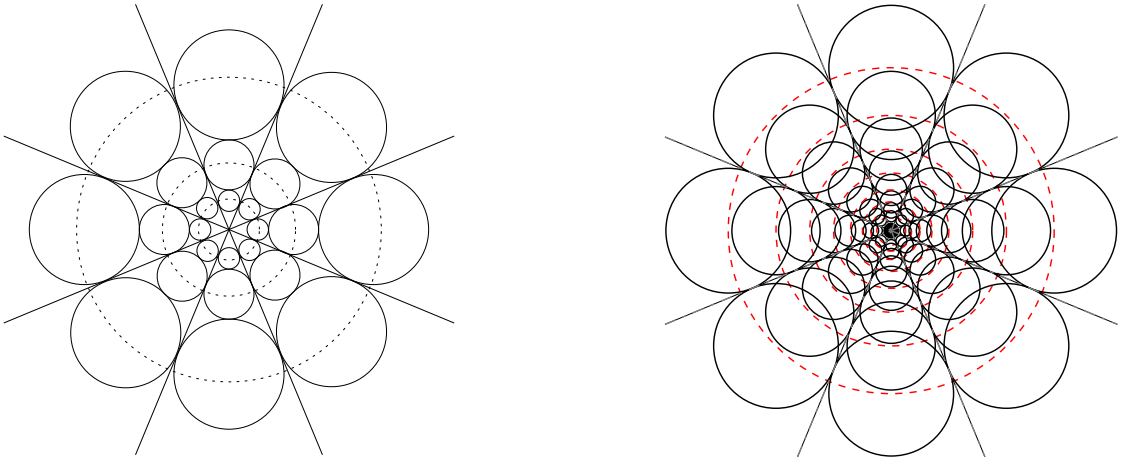
[7]The Gaussian filtering is actually used for a scale i only if $\sigma_i > 0.6$, as filtering with lower σ_i leads to significant loss of information.

Similarly to [59], the filters are designed so that 95% of their mass lies within an effective area of radius r_i. We select the effective area diameter, such that the effective areas at the same scale touch each other:

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

The proportion of normal distribution mass in z standard deviations is equal to $\mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)$, where erf denotes the Gauss error function. There from the standard deviation σ_i for the Gaussian filter at scale i can be expressed as:

$$\sigma_i = \frac{r_i}{z} = \frac{r_i}{\sqrt{2} \cdot \mathrm{erf}^{-1}(0.95)} \approx \frac{r_i}{1.959964} \tag{9}$$



(a) Scale space by Mäenpää [59] (b) The scale space proposed in FSRIT

Figure 6: The effective areas of filtered pixel samples in a multi-resolution $\mathrm{LBP}_{8,R}$ operator

Having the obtained scale space, LBP-HF-S-M histograms from c adjacent scales are concatenated into a single descriptor. Invariance to scale changes is increased by creating n_{conc} multi-scale descriptors for one image.

In the experiments in Section 4, $n_{conc} = 3$ multi-scale descriptors are used, each consisting of $c = 6$ scales of LBP-HF-S-M features. The effect of parameters c (i.e. the length of multi-scale description) and $n_{conc} = 3$ (i.e. the robustness to scale change) is illustrated in Section 4.

See Algorithm 1 for the overview of the texture description method.

Algorithm 1 The FSRIT description method overview

$R_1 := 1$
for all scales $i = 1...(n_{conc} + c - 1)$ **do**
  $\sigma_i := R_i \sin \frac{\pi}{P} / 1.959964$
  if $\sigma_i > 0.6$ **then**
    apply Gaussian filter (with std. dev. σ_i) on the original image
  end if
  extract LBP_{P,R_i}-S and LBP_{P,R_i}-M and build the LBP-HF-S-M descriptor
  for $j = 1...c$ **do**
    if $i \geq j$ and $i < j + n_{conc}$ **then**
      attach the LBP-HF-S-M to the j-th multi-scale descriptor
    end if
  end for
  $R_{i+1} := R_i \sqrt{2}$
end for
apply an approx. χ^2 kernel map on the concatenated multi-scale descriptor

3.5 Support Vector Machine and feature maps

In common applications, a Support Vector Machine (SVM) classifier with non-linear kernel usually provides higher recognition accuracy at the price of significantly higher time complexity and higher storage demands (dependent on the size of training set). An approach for efficient additive kernels via explicit feature maps is described by Vedaldi and Zisserman [64] and can be combined with linear SVM classifiers. Using linear SVMs on feature mapped data improves the recognition accuracy, while preserving the linear SVM advantages like fast evaluation and low storage load (independent on the number of training samples), which are both very practical in real time applications.

In our experiments we use a Stochastic Dual Coordinate Ascent [65] linear SVM solver implemented in the VLFeat library [66].

In FSRIT we make use of the explicit feature map approximation of the χ^2 kernel, though in some cases the histogram intersection kernel map may give even better results, as discussed in Section 4. In common applications, the data are only trained once and the training precision is more important than the training time. Thus in our SVM parameter setting, we demand high accuracy: $\sigma = 10^{-7}$, $\epsilon = 10^{-7}$, max. # of iterations: 10^8.

The "One versus All" classification scheme is used for multi-class classification, implementing the Platt's probabilistic output [67, 68] to ensure SVM results comparability among classes. The maximal posterior probability estimate over all scales is used to determine the resulting class.

4 Experiments

In all following experiments, we use the same setting of our method: $n_{\text{conc}} = 3$ multi-scale descriptors per image are used, each of them consisting of $c = 6$ scales described using LBP-HF-S-M. The final histogram is kernelized using the χ^2 kernel map. We use the unified setting in order to show the generality of FSRIT description, although setting the parameter individually for a given dataset can further increase the accuracy. Figures 7 and 8 illustrate the effect of different parameter settings on the recognition accuracy for the KTH-TIPS2b texture database.

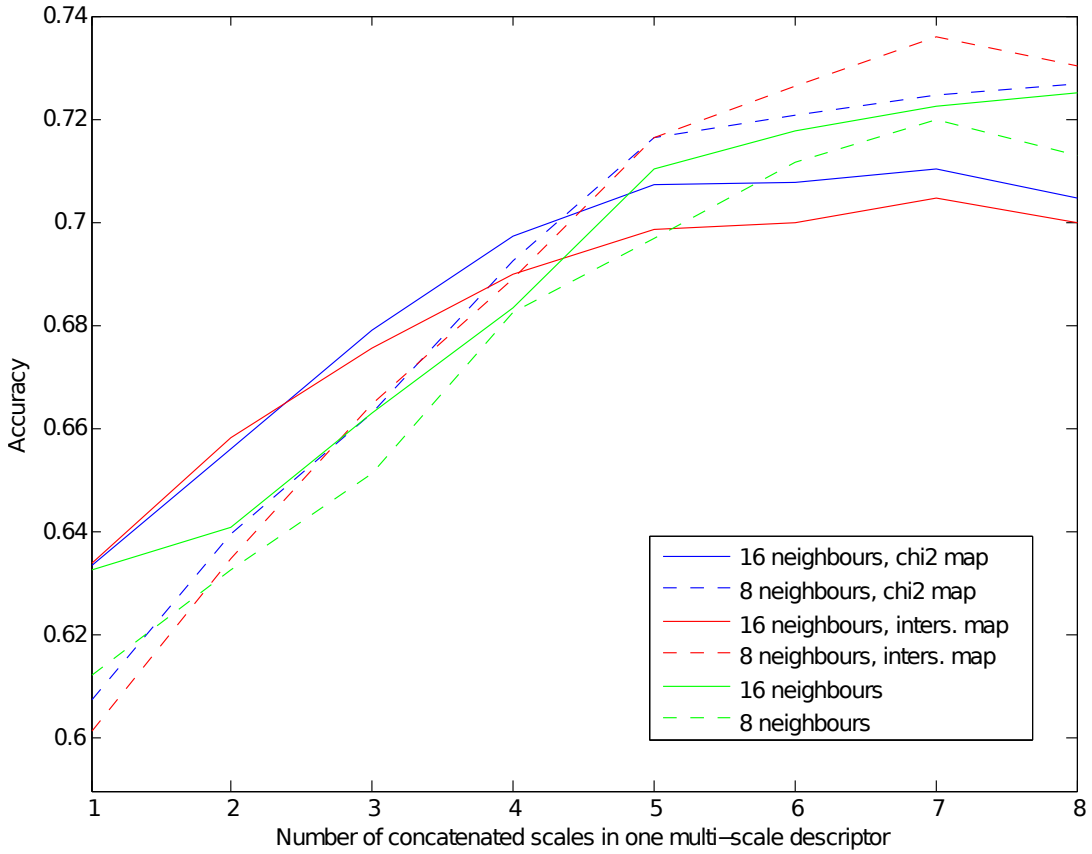


Figure 7: Feature mapping and concatenating features from multiple scales in FSRIT, KTH-TIPS2b

To reduce the effect of random training and testing data choice, the presented results are averaged from 10 experiments.

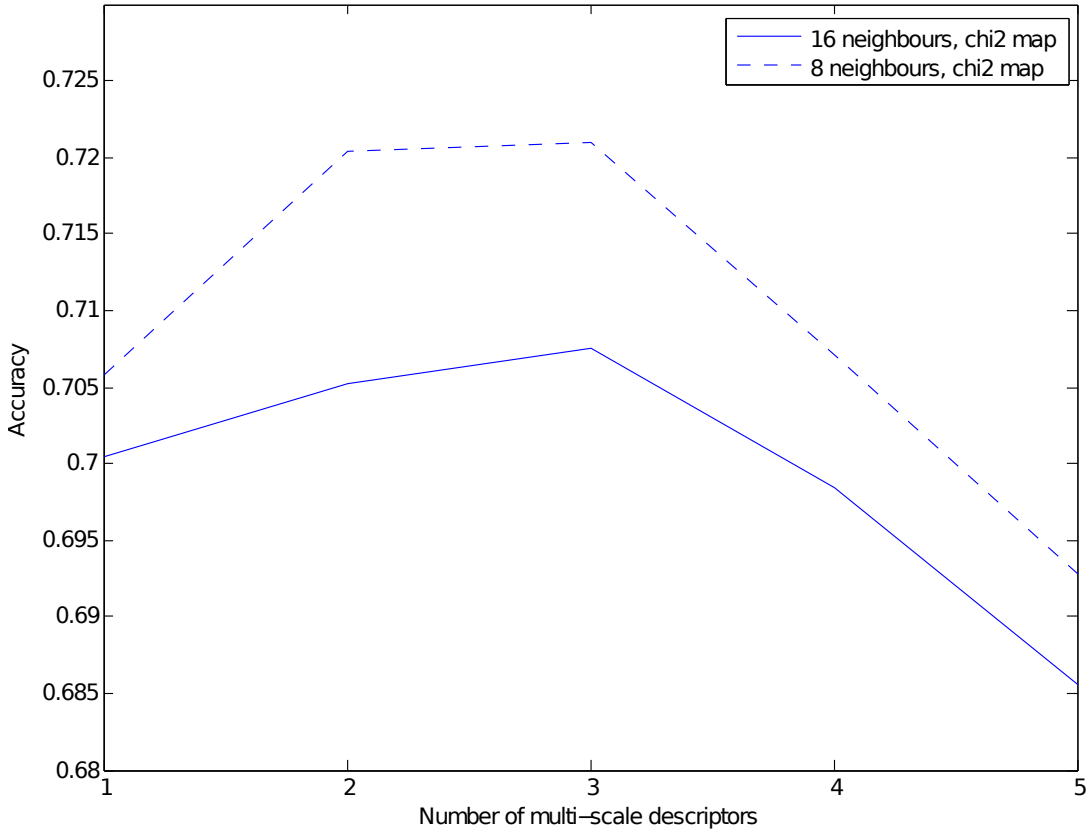


Figure 8: Dependency on the number c of multiscale descriptors in FSRIT, KTH-TIPS2b

4.1 Texture classification

To evaluate the accuracy of FSRIT, it was tested on the following texture datasets and the results were compared to state-of-the-art texture classification results (Tables 1 and 2).

4.1.1 Brodatz32

The Brodatz32 dataset [69] was published in 1998, and it contains low resolution (64x64 px) grey-scale images of 32 textures from the photographs published by Phil Brodatz [70] in 1966, with artificially added rotation (90°) and scale change (a 64x64 px scaled block obtained from 45x45 pixels in the middle). Totally there are 64 images for each texture class.

The standard protocol for this dataset simply divides the data into two halves (i.e. 32 images per class in the training set and 32 in the testing set).

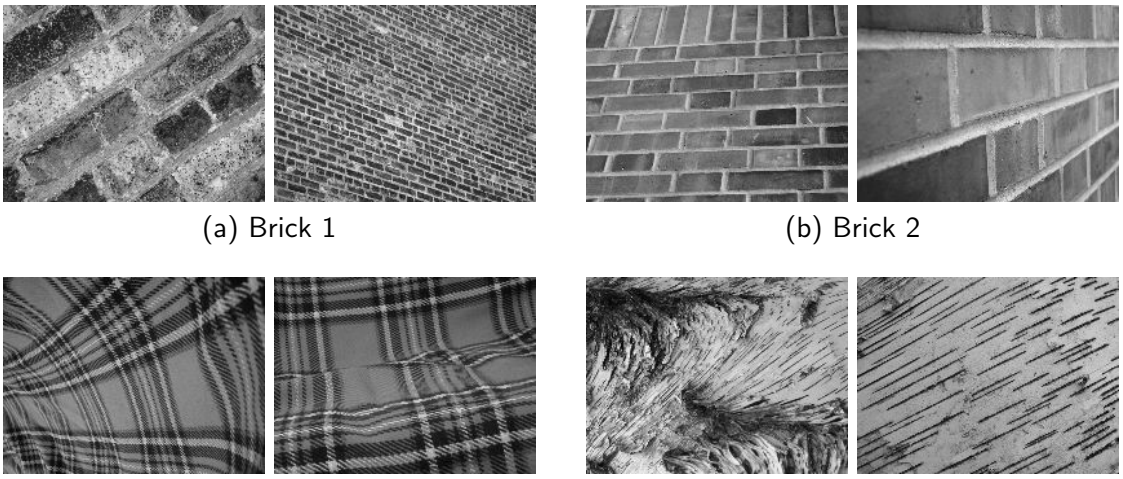
Even though the original images are copyrighted and the legality of their usage in

academic publications is unclear[8], Brodatz textures are one of the most popular and broadly used sets in texture analysis.

4.1.2 UIUCTex

The Ponce Group Texture Database, which is usually referred to as UIUCTex, was published by Lazebnik et al. [71] in 2005 and features 25 different texture classes, 40 samples each. All images are in VGA resolution (640x480 px) and in grey-scale.

The surfaces included in the database are of various nature (wood, marble, gravel, fur, carpet, brick, ..) and were acquired with significant viewpoint, scale and illumination changes and additional sources of variability, including, but not limited to, non-rigid material deformations (fur, fabric, and water) and viewpoint-dependent appearance variations (glass). Examples of images from different classes are in Figure 9.



(a) Brick 1 (b) Brick 2

(c) Plaid (d) Bark 3

Figure 9: Examples of 4 texture classes from the UIUCTex database

The results on this dataset are usually evaluated using 20 or 10 training images per class. In our experiments, the former case with a larger training set is performed.

4.1.3 KTH-TIPS

The KTH-TIPS texture database [72, 73] contains images of 10 materials. There are 81 images (200x200 px) of each material with different combination of pose, illumination and scale.

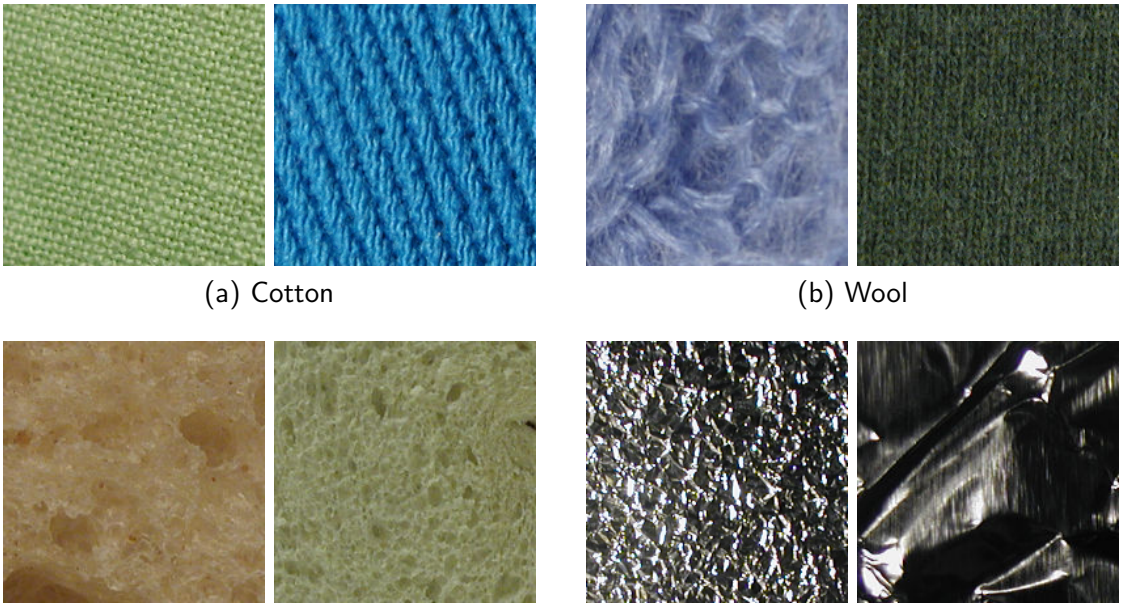
The standard evaluation scheme on this dataset uses 40 training images per material.

[8] `http://graphics.stanford.edu/projects/texture/faq/brodatz.html`

4.1.4 KTH-TIPS2

The KTH-TIPS2 database was published [74, 75] shortly after KTH-TIPS. It builds on the KTH-TIPS database, but provides multiple sets of images - denoted as samples - per material class (examples in Figure 10).

There are 4 samples for each of the 11 materials in the KTH-TIPS2 database, containing 108 images per sample (again with different combination of pose, illumination and scale). However, in the first version of this dataset, for 4 of those 44 samples only 72 images were used. This first version is usually denoted as KTH-TIPSa, and the standard evaluation method uses 3 samples from each class for training and 1 for testing. The "complete" version of this database, KTH-TIPSb, is usually trained only on 1 sample per class and tested on the remaining 3 samples.



(a) Cotton (b) Wool

(c) White bread (d) Aluminium foil

Figure 10: Examples of 4 texture classes from the KTH-TIPS2 database

4.1.5 CUReT

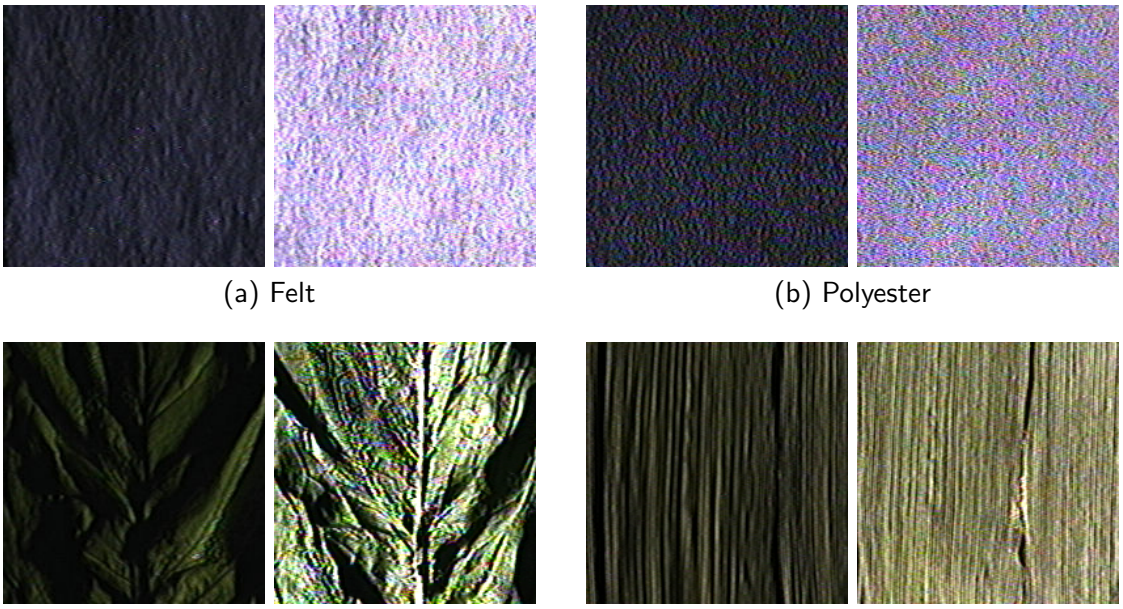
The CUReT [76] image database contains textures from 61 classes, each observed with 205 different combinations of viewing and illumination directions, i.e. 12505 images in total. In the commonly used version of the dataset, all images have a resolution of 200x200 px.

Though CUReT also contains a BRDF (bidirectional reflectance distribution function)

Method	*KTH-TIPS2a*	*KTH-TIPS2b*	*KTH-TIPS*
FSRIT_8, $\text{K}\chi^2$	**86.2±5.5**	**72.1±5.1**	98.9±0.7
FSRIT_{16}, $\text{K}\chi^2$	84.6±5.8	70.8±2.9	98.8±0.8
IFV_{SIFT} [55]	82.5±5.2	69.3±1.0	**99.7±0.1**
Scattering [53]	–	–	99.4±0.4
LHS [47]	73.0±4.7	–	–
SR-EMD-M [54]	–	–	**99.8**

Table 1: Evaluation of FSRIT on the KTH-TIPS datasets, compared to the state-of-the-art methods

database, for purposes of standard texture recognition methods, only the image (resp. bidirectional texture function) database is used. We use 46 training images per class, which is a standard evaluation scheme for the CUReT database.



(a) Felt (b) Polyester

(c) Lettuce Leaf (d) Corn Husk

Figure 11: Examples of 4 texture classes from the CUReT database

4.1.6 UMD

The UMD dataset [77] consists of 1000 uncalibrated, unregistered grey-scale images of size 1280x960 px, 40 images for each of 25 different textures. The UMD database contains non-traditional textures like images of fruits, shelves of bottles and buckets, various plants,

or floor textures.

The standard evaluation scheme for UMD is dividing the data into two halves (i.e. 20 images per class in the training set and 20 in the testing set).

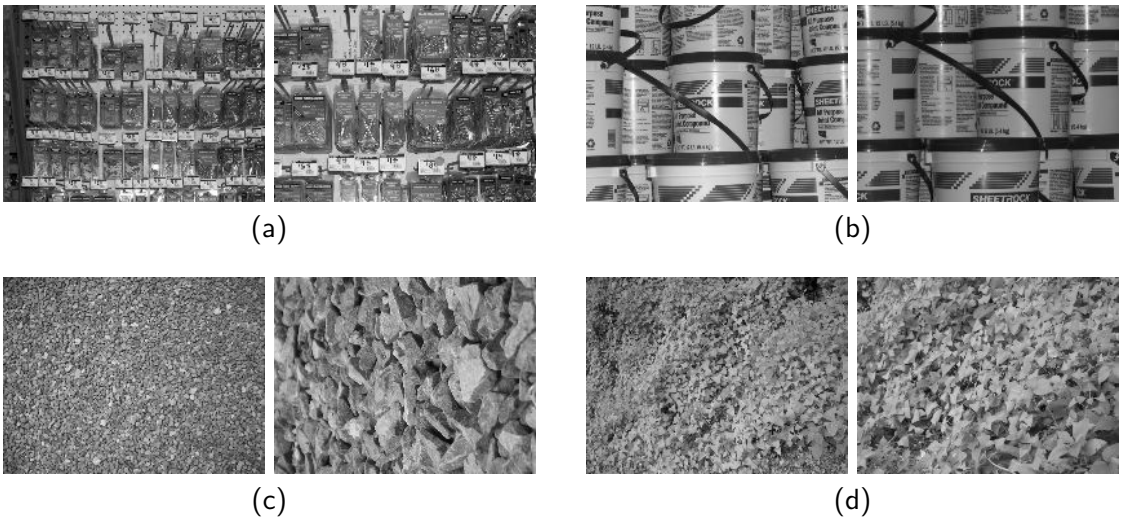


Figure 12: Examples of 4 texture classes from the UMD database

Method	*Brodatz32*	*UIUCTex*	*UMD*	*CUReT*
$FSRIT_8$, $K\chi^2$	99.2±0.3	98.6±0.6	99.3±0.3	98.5±0.2
$FSRIT_{16}$, $K\chi^2$	**99.7±0.3**	99.3±0.3	99.3±0.4	98.8±0.2
IFV_{SIFT} [55]	–	–	99.2±0.4	**99.6±0.3**
Scattering [53]	–	**99.4±0.4**	**99.7±0.3**	–
LHS [47]	99.5±0.2	–	–	–
SR-EMD-M [54]	–	–	**99.9**	99.5

Table 2: Evaluation of FSRIT on other standard datasets, compared to the state-of-the-art methods

4.2 Suitability for real-time applications

Table 3 shows comparison of our image processing times to the state-of-the-art texture recognition method by Cimpoi et al. [55] using IVF_{SIFT}. Both the implementation of FSRIT and IVF_{SIFT}[9] used MATLAB scripts with a C code in the VLFEAT [66] framework (where we added a new CLBP implementation for our method). The processing times were measured on a standard laptop (1.3 GHz Intel Core i5, 4 GB 1600 MHz DDR3) without

[9]Using the code kindly provided by the authors of [55]

parallelization.

Method	*KTH-TIPS2b*	*KTH-TIPS*	*CUReT*
FSRIT_{16}, $\text{K}\chi^2$	0.059 s / im.	0.057 s / im.	0.025 s / im.
FSRIT_{8}, $\text{K}\chi^2$	0.038 s / im.	0.032 s / im.	0.014 s / im.
IFV_{SIFT} [55]	0.106 s / im.	0.149 s / im.	0.055 s / im.

Table 3: Average image description time on selected datasets, compared to IFV_{SIFT}

The average description time for a low resolution (0.4 Mpx) image for FSRIT_8 is at most 0.04 s, while for higher resolutions the processing time will grow proportional to the image resolution, as the number of local operations will increase with the number of pixels.

4.3 Recognition of leaves and bark

The following experiments are focused on the application of texture recognition using the FSRIT method for recognition of trees and show that the method is suitable both for recognition of tree bark and leaves. The parameter setting is the same as in the previous experiments, the only difference is that the leaf images are segmented (simply by thresholding using the Otsu's method [9]) and the texture statistics are computed only from the leaf area.

4.3.1 Austrian Federal Forest (AFF) datasets

Fiel and Sablatnig [7] used the Österreichische Bundesforste – Austrian Federal Forests (AFF) datasets for recognition of trees based on images of its leaves, bark and needles. Although the datasets are not publicly available, the Computer Vision Lab, TU Vienna, kindly made them available to us for academic purposes, with courtesy by Österreichische Bundesforste/Archiv. The AFF dataset of bark contains 1182 bark images from 11 classes (tree species). The AFF dataset of leaves contains 134 photos of leaves (on white background) of the 5 most common Austrian broad leaf trees.

The results comparison using the Fiel-Sablatnig protocol, i.e. using 8 training images per leaf class and 15 training images per bark class, is in Table 4. The results using 10-fold cross validation in Table 5 show that the accuracy increases significantly using a larger training set.

Figure 13: Examples from the AFF bark dataset

Method	*Bark*	*Leaves*
$FSRIT_8$, $K\chi^2$	80.8±2.3	**96.5±1.2**
$FSIT_8^{u2}$, $K\chi^2$	**82.0±2.2**	96.3±1.6
SIFT (Bag of Words) [7]	–	93.6
+ GCLM + wavelet features [7]	69.7	–
MB-LBP [10] reimplemented in [15]	51.2±2.9	

Table 4: Evaluation of FSRIT on the AFF datasets of leaves and bark using the Fiel-Sablatnig protocol

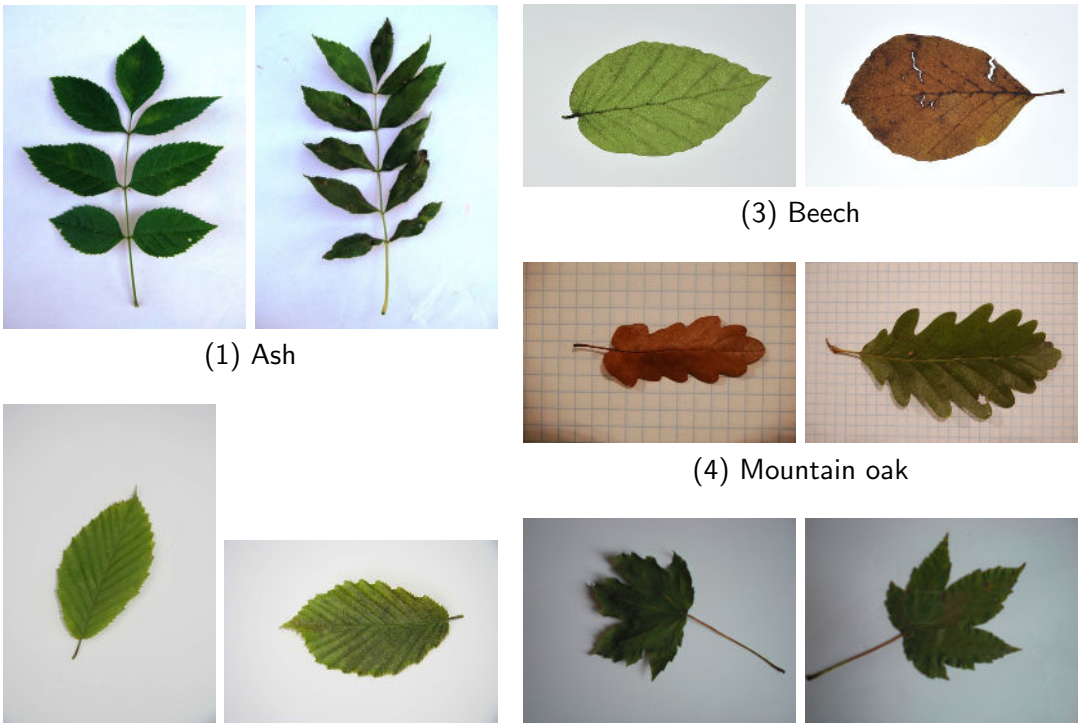


Figure 14: Examples from the AFF leaf dataset

Method	*Bark*	*Leaves*
FSRIT_8, $\text{K}\chi^2$	93.6±1.9	97.3±4.3
$\text{FSIT}_8^{\text{u2}}$, $\text{K}\chi^2$	96.6±2.0	98.2±3.8

Table 5: Evaluation of FSRIT on the AFF datasets of leaves and bark using 10 fold cross validation

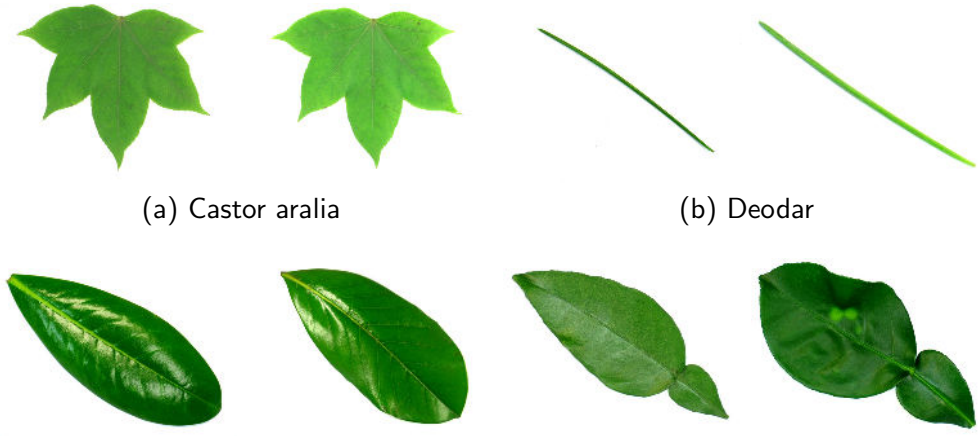
4.3.2 Flavia leaf dataset

The Flavia dataset contains 1907 images (1600x1200 px) of leaves from 32 plant species on white background, 50 to 77 images per class.

Even though in the original paper by Wu et al. [1] 10 images per class are used for testing and rest of the images for training, most recent publications use 10 randomly selected testing images and 40 randomly selected training images per class, achieving better recognition accuracy even with the lower number of training samples. In the case of the two best result reported by Lee et al. [20, 21], the number of training samples is not clearly stated[10]. Some papers divide the set of images for each class into two halves, one

[10]Especially in [20] the result presented as "95.44% (1820 / 1907)" seems to be tested on all images

being used for training and the other for testing. The comparison of results using both schemes is in Table 6.



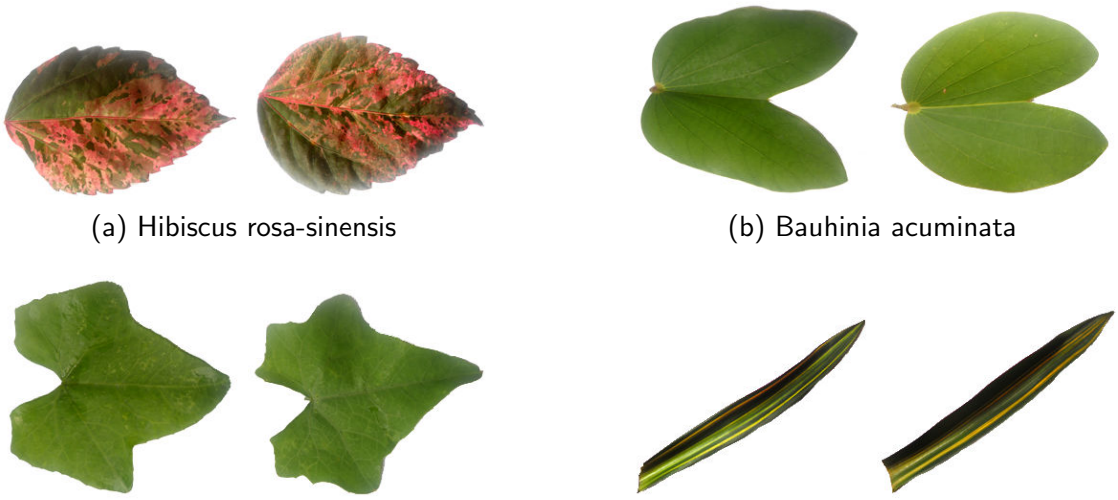
(a) Castor aralia (b) Deodar

(c) Southern magnolia (d) Tangerine

Figure 15: Examples of 4 classes from the Flavia leaf dataset

4.3.3 Foliage leaf dataset

The Foliage leaf dataset [19, 24] contains 60 classes of leaves from 58 species. The dataset is already divided into a training set with 100 images per class and a testing set with 20 images per class. The accuracy using this evaluation protocol is compared to the best reported results in Table 6.



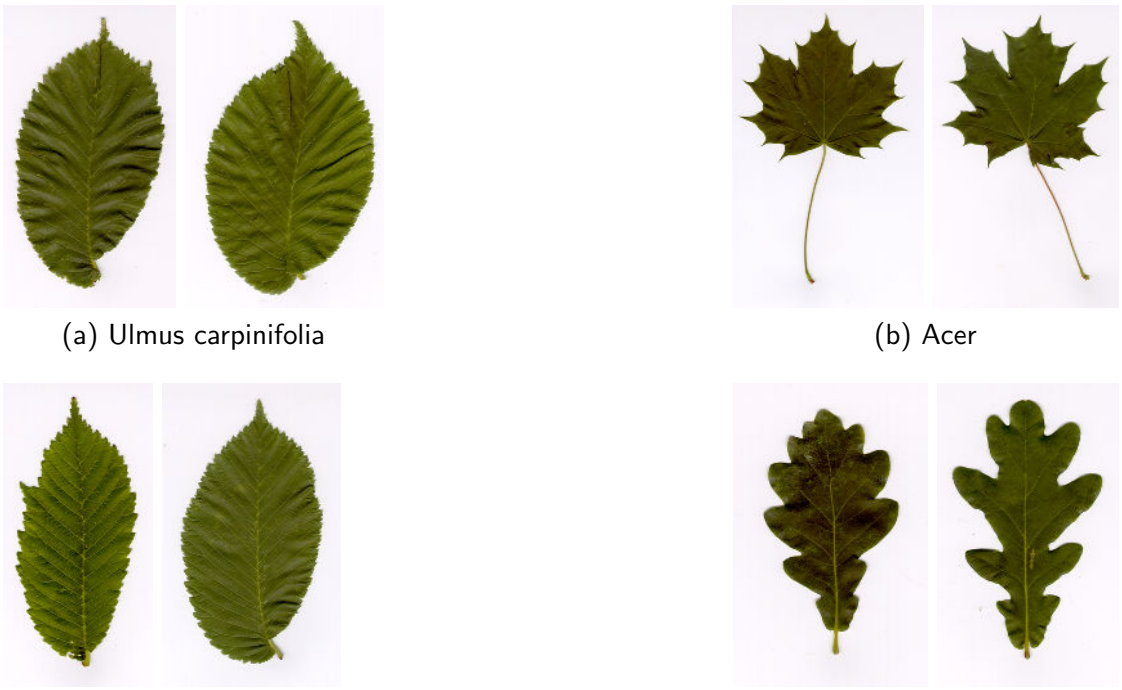
(a) Hibiscus rosa-sinensis (b) Bauhinia acuminata

(c) Ipomoea lacunose (d) Tradescantia spathacea "Vittata"

Figure 16: Examples of 4 classes from the Foliage dataset

4.3.4 Swedish leaf dataset

The Swedish leaf dataset was introduced in the Söderkvist's diploma thesis [2] and contains images of leaves scanned using 300 dpi colour scanner. There are 75 images for each of 15 contained tree classes. The standard evaluation scheme uses 25 images for training and the remaining 50 for testing. Our results on the Swedish dataset are included in Table 6.



(a) Ulmus carpinifolia (b) Acer

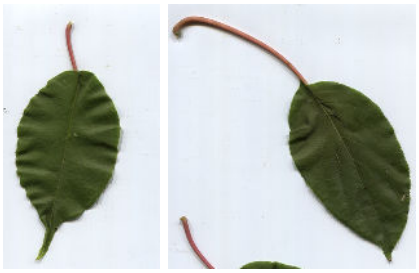
(c) Salix aurita (d) Quercus

Figure 17: Examples of 4 classes from the Swedish dataset

4.3.5 Middle European Woods (MEW) dataset

The MEW dataset recently introduced by Novotný and Suk [22] contains 300 dpi scans of leaves belonging to 153 classes of Central European trees and shrubs. There are 9745 samples in total, at least 50 per class. Our experiment is performed using half of the images in each class for training and the other half for testing and the results in Table 6 are compared to the authors' using the same evaluation scheme.

(a) Acer campestre



(b) Actinidia arguta

(c) Berberis thunbergii

(d) Zelkova serrata

Figure 18: Examples of 4 classes from the MEW dataset

Method	*Flavia* 10×40	*Flavia* $\frac{1}{2} \times \frac{1}{2}$	Foliage	Swedish	MEW
FSRIT_8, $\text{K}\chi^2$	98.3±0.6	**97.8±0.4**	**96.17**	**99.6±0.3**	96.8±0.3
$\text{FSIT}_8^{\text{u2}}$, $\text{K}\chi^2$	**98.6±0.6**	98.1±0.3	95.25	99.6±0.5	**97.2±0.3**
Fourier descriptors [22]	–	91.5	–	–	84.9
Circular Shift Method [23]	–	96.5	–	–	–
IDSC+CCD [10]	–	83.3	–	–	–
Shape, vein, texture and colour features + PCA + PNN [18]	95.0	–	95.8	–	–
Zernike moments of different features + PNN [19]	94.7	–	93.3	–	–
Methods by Lee et al. [20, 21][a]	97.2, 95.4	–	–	–	–
Spatial PACT [25]	–	–	–	97.9	–
PRI-CoLBP$_g$ + SVM [26][b]	–	–	–	99.4	–

Table 6: Evaluation of FSRIT on the leaf datasets (Flavia, Foliage, Swedish, Middle European Woods)

[a] the evaluation scheme in this papers is not clearly described, as discussed in Section 4.3.2

[b] according to the project homepage `http://qixianbiao.github.io`

4.4 Combining leaf and bark

For both the description of bark and leaf, the probabilistic outputs of SVM classifiers are computed, as discussed in Section 3.5. Thus for i-th class c_i we have the estimates of posterior probabilities $P(c_i|y_{Bi})$ and $P(c_i|y_{Li})$, where y_{Bi} and y_{Li} denote the SVM outputs for bark and leaf samples before the sigmoid mapping. Under the assumption that the observations from leaf and bark are class-conditionally independent, we may compute the probability of belonging to a given class c_i given the observations as:

$$P(c_i|y_{Bi}, y_{Li}) = \frac{P(y_{Bi}, y_{Li}|c_i) \cdot P(c_i)}{P(y_{Bi}, y_{Li})} = \frac{P(y_{Bi}|c_i) \cdot P(y_{Li}|c_i) \cdot P(c_i)}{P(y_{Bi}, y_{Li})} =$$

$$= \frac{\frac{P(c_i|y_{Bi}) \cdot P(y_{Bi})}{P(c_i)} \cdot \frac{P(c_i|y_{Li}) \cdot P(y_{Li})}{P(c_i)} \cdot P(c_i)}{P(y_{Bi}, y_{Li})} = \tag{10}$$

$$= \frac{P(y_{Bi}) \cdot P(y_{Li})}{P(y_{Bi}, y_{Li})} \cdot \frac{P(c_i|y_{Bi}) \cdot P(c_i|y_{Li})}{P(c_i)} = k \cdot \frac{P(c_i|y_{Bi}) \cdot P(c_i|y_{Li})}{P(c_i)}$$

where k is class independent. Assuming that all class prior probabilities $P(c_i)$ are equal, we end up with a simple product rule, where the resulting class is found as follows:

$$c_{\prod} = \arg\max_i \left(P(c_i|y_{Bi}) \cdot P(c_i|y_{Li})\right) \tag{11}$$

The experimental results of Kittler et al. [78] show that combining classifiers using a simple sum rule can lead to a better accuracy than using the product rule. We thus also use a sum rule for comparison:

$$c_{\sum} = \arg\max_i \left(P(c_i|y_{Bi}) + P(c_i|y_{Li})\right) \tag{12}$$

Our experiment was performed on pairs of bark and leaf images from the AFF databases. Five classes common to the AFF bark dataset and AFF leaf dataset were combined, each containing at least 16 images of each kind. From each class, we have randomly selected 8 bark and 8 leaf samples for training and another 8 pairs of bark and leaf for testing. The results of this experiment in Table 7 show that both proposed rules of combining the classifiers lead to improvement in accuracy, the sum rule performing slightly better.

Method	*Bark*	*Leaves*	*Combination*	
			$\prod$	$\sum$
FSRIT_8, $\text{K}\chi^2$	90.0±6.0	96.0±2.7	97.5±2.9	**98.0±2.0**
$\text{FSIT}_8^{\text{u2}}$, $\text{K}\chi^2$	88.8±6.9	97.8±2.2	98.8±1.8	**99.0±1.3**

Table 7: Evaluation of FSRIT on the AFF datasets, combining both leaves and bark classification

4.5 Species retrieval and classification errors

As discussed in Section 1, even results which are not correctly classified may be useful, if the correct species appears in a shortlist (i.e. among the several first returned results).

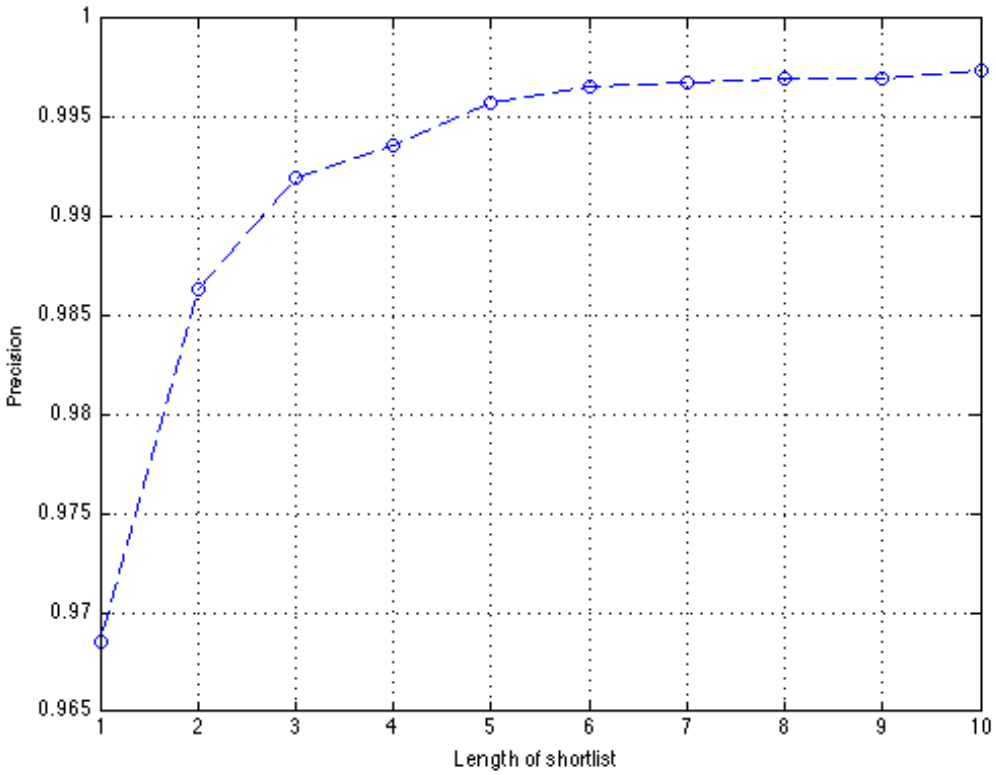


Figure 19: Retrieval precision for different lengths of shortlist, MEW leaf dataset (153 classes)

Figures 19 and 20 illustrate the precision of our method in retrieval on the MEW leaf database (153 classes, using half of the images for training and half for testing) and AFF bark database (11 classes, using 10-fold cross validation).

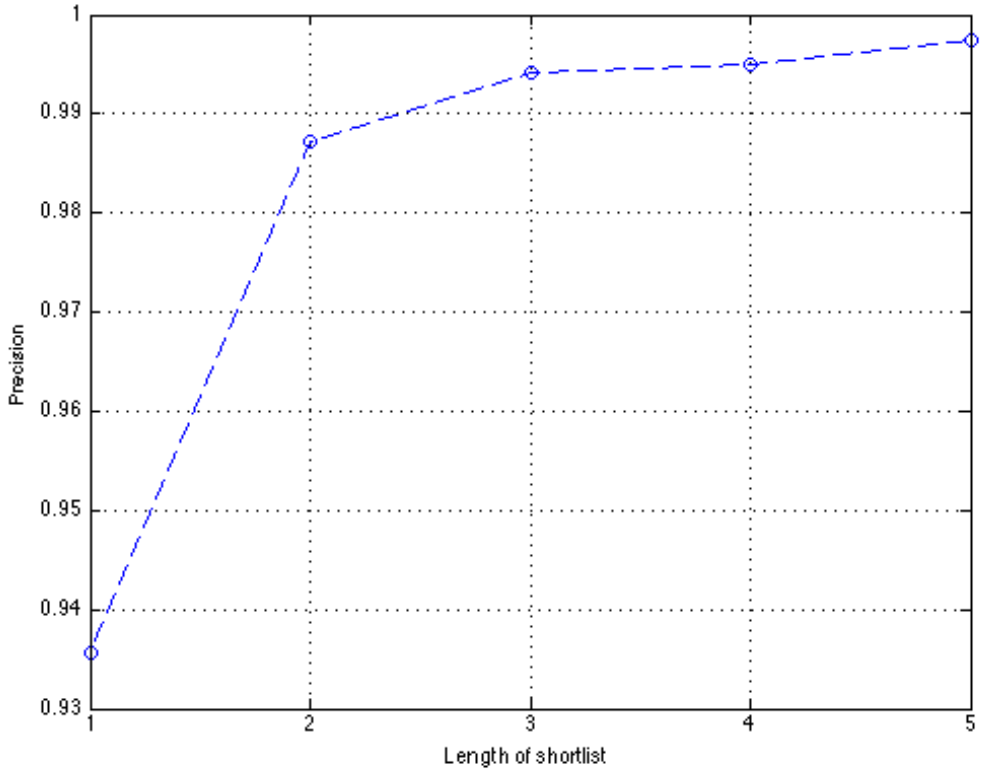


Figure 20: Retrieval precision for different lengths of shortlist, AFF bark dataset (11 species)

Figures 21 and 22 in Appendix C contain examples of misclassification on the AFF bark dataset and MEW leaf dataset. For illustration, the images are in displayed grey-scale, as processed by the FSRIT method. In the examples of leaf misclassification, the segmented background is represented by blue color.

5 Conclusions

This thesis has several contributions:

We proposed to approach the problems of bark and leaf identification from images as texture recognition problems. In order to obtain a suitable and robust texture description which will be obtained and recognized quickly, we proposed a novel texture recognition method. This Fast Scale and Rotation Invariant Texture method (FSRIT) is based on multi-scale texture description using concatenated rotation invariant features obtained from uniform LBP histograms computed at multiple scales. Robustness to scale changes is handled by creating multiple descriptors concatenated from different scales. A linear SVM classifier was chosen for its fast evaluation and low memory footprint. To improve recognition accuracy, the Support Vector Machines was learned and evaluated on data "kernelized" using feature map approximating the χ^2 kernel.

Experimental results show that the proposed method outperforms the state of the art for both bark recognition and leaf recognition, as well as the best reported results on the most challenging standard texture datasets.

In the bark recognition problem we achieved 93.6% accuracy on the AFF bark dataset with 10-fold cross validation (respectively 96.6% accuracy using rotation-dependent description, which can be used in applications like a mobile field guide, where it is simple for the user to take the picture of tree trunk in "up-is-up" orientation). Even experiments using only 15 training samples per class achieved 80.8% accuracy (respectively 82.0% using rotation-dependent description), which outperforms the best reported results, as well as accuracy of both human experts from the Austrian Federal Forests in bark recognition experiments [7].

In the leaf recognition problem, we exceeded the best reported results on all tested datasets. The average accuracies obtained with the rotation invariant description were 98.3% on the Flavia dataset, 96.2% on the Foilage dataset, 99.6% on the Swedish dataset and 96.2% on the Middle European Woods (MEW) dataset.

In terms of retrieval the correct species is obtained in the top 3 results in more than 99% cases on the MEW leaf dataset as well as on the AFF bark dataset.

In addition show that combining both the observations of bark and leaf can further improve the recognition. Both the product rule and sum rule for combining the bark and leaf classifiers improved the accuracy, the later achieving slightly better results.

Experiments using the proposed FSRIT method with the same setting also proved

its suitability for general texture problems. Our average accuracies on the KTH-TIPS2 databases (86% for KTH-TIPS2a and 72% for KTH-TIPS2b) are higher than the best reported results, while the method achieves 99% accuracy results on other standard texture datasets (KTH-TIPS, CUReT, UIUCTex, UMD and Brodatz32).

We have shown that the proposed recognition method is very fast and suitable for real-time applications, including (but not limited to) implementation on mobile devices, e.g. to create an intelligent mobile field guide.

5.1 Future work

Because the FSRIT description is performed on grey-scale images, combining it with some colour features might further improve the recognition accuracy in cases, where the colour is a distinctive attribute.

The leaf recognition accuracy might also be improved by combining with shape features, as we can observe, that some misclassified leaf samples (in Figure 22, Appendix C) might be similar in texture, but different in shape.

Description of high resolution images could be further sped up by computing the CLBP features for only a subset of points, obtained by pseudo-random sampling. An analysis comparing the dependency of accuracy on the number of sampled points would be needed.

References

[1] Stephen Gang Wu, Forrest Sheng Bao, Eric You Xu, Yu-Xuan Wang, Yi-Fan Chang, and Qiao-Liang Xiang. A leaf recognition algorithm for plant classification using probabilistic neural network. In *Signal Processing and Information Technology, 2007 IEEE International Symposium on*, pages 11–16. IEEE, 2007.

[2] Oskar Söderkvist. Computer vision classification of leaves from swedish trees. 2001.

[3] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *PAMI*, 24(7):971–987, 2002.

[4] Timo Ahonen, Jiří Matas, Chu He, and Matti Pietikäinen. Rotation invariant image description with local binary pattern histogram fourier features. In *SCIA '09, in Proc.*, pages 61–70. Springer-Verlag, 2009.

[5] Peter N Belhumeur, Daozheng Chen, Steven Feiner, David W Jacobs, W John Kress, Haibin Ling, Ida Lopez, Ravi Ramamoorthi, Sameer Sheorey, Sean White, and Ling Zhang. Searching the world's herbaria: A system for visual identification of plant species. In *Computer Vision–ECCV 2008*, pages 116–129. Springer, 2008.

[6] Maria-Elena Nilsback and A Zisserman. *An Automatic Visual Flora: Segmentation and Classification of Flower Images*. PhD thesis, Oxford University, 2009.

[7] Stefan Fiel and Robert Sablatnig. Automated identification of tree species from images of the bark, leaves and needles. In *Proc. of 16th Computer Vision Winter Workshop*, pages 1–6, Mitterberg, Austria, 2011.

[8] Neeraj Kumar, Peter N Belhumeur, Arijit Biswas, David W Jacobs, W John Kress, Ida C Lopez, and João VB Soares. Leafsnap: A computer vision system for automatic plant species identification. In *Computer Vision–ECCV 2012*, pages 502–516. Springer, 2012.

[9] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

[10] Tomáš Sixta and Jiří Matas. Image and video-based recognition of natural objects. Diploma thesis, Czech Technical University in Prague, 2011.

[11] Zheru Chi, Li Houqiang, and Wang Chao. Plant species recognition based on bark patterns using novel gabor filter banks. In *ICNNSP, in Proc.*, volume 2, 2003.

[12] Yuan-Yuan Wan, Ji-Xiang Du, De-Shuang Huang, Zheru Chi, Yiu-Ming Cheung, Xiao-Feng Wang, and Guo-Jun Zhang. Bark texture feature extraction based on statistical texture analysis. In *ISIMP, in Proc.*, 2004.

[13] Jiatao Song, Zheru Chi, Jilin Liu, and Hong Fu. Bark classification by combining grayscale and binary texture features. In *ISIMP, in Proc.*, 2004.

[14] Zhi-Kai Huang, Chun-Hou Zheng, Ji-Xiang Du, and Yuan-yuan Wan. Bark classification based on textural features using artificial neural networks. In *Advances in Neural Networks-ISNN'2006*. Springer, 2006.

[15] M. Sulc and J. Matas. Kernel-mapped histograms of multi-scale lbps for tree bark recognition. In *Image and Vision Computing New Zealand (IVCNZ), 2013 28th International Conference of*, pages 82–87, Nov 2013.

[16] Abdul Kadir, Lukito Edi Nugroho, Adhi Susanto, and Paulus Insap Santosa. A comparative experiment of several shape methods in recognizing plants. *International Journal of Computer Science & Information Technology*, 3(3), 2011.

[17] Gaurav Agarwal, Peter Belhumeur, Steven Feiner, David Jacobs, W John Kress, Ravi Ramamoorthi, Norman A Bourg, Nandan Dixit, Haibin Ling, Dhruv Mahajan, et al. First steps toward an electronic field guide for plants. *Taxon*, 55(3):597–610, 2006.

[18] Abdul Kadir, Lukito Edi Nugroho, Adhi Susanto, and Paulus Insap Santosa. Performance improvement of leaf identification system using principal component analysis. *International Journal of Advanced Science & Technology*, 44, 2012.

[19] Abdul Kadir, Lukito Edi Nugroho, and P Insap Santosa. Experiments of zernike moments for leaf identification 1. 2012.

[20] Kue-Bum Lee and Kwang-Seok Hong. Advanced leaf recognition based on leaf contour and centroid for plant classification. In *The 2012 International Conference on Information Science and Technology*, pages 133–135, 2012.

[21] Kue-Bum Lee, Kwang-Woo Chung, and Kwang-Seok Hong. An implementation of leaf recognition system, 2013.

[22] Petr Novotný and Tomáš Suk. Leaf recognition of woody species in central europe. *Biosystems Engineering*, 115(4):444–452, 2013.

[23] G Karuna, B Sujatha, Rajahumndry GIET, and P Chandrasekhar Reddy. An efficient representation of shape for object recognition and classification using circular shift method.

[24] Abdul Kadir, Lukito Edi Nugroho, Adhi Susanto, and Paulus Insap Santosa. Neural network application on foliage plant identification. *International Journal of Computer Applications*, 29, 2011.

[25] Jianxin Wu and Jim M Rehg. Centrist: A visual descriptor for scene categorization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(8):1489–1501, 2011.

[26] Xianbiao Qi, Rong Xiao, Jun Guo, and Lei Zhang. Pairwise rotation invariant co-occurrence local binary pattern. In *Computer Vision–ECCV 2012*, pages 158–171. Springer, 2012.

[27] R Pydipati, TF Burks, and WS Lee. Identification of citrus disease using color texture features and discriminant analysis. *Computers and electronics in agriculture*, 52(1):49–59, 2006.

[28] Seon-Jong Kim, Byeong-Wan Kim, and Dong-Pil Kim. Tree recognition for landscape using by combination of features of its leaf, flower and bark. In *SICE Annual Conference (SICE), 2011 Proceedings of*, 2011.

[29] Jung-Hyun Kim, Rong-Guo Huang, Sang-Hyeon Jin, and Kwang-Seok Hong. Mobile-based flower recognition system. In *Intelligent Information Technology Application, 2009. IITA 2009. Third International Symposium on*, volume 3, pages 580–583. IEEE, 2009.

[30] Takeshi Saitoh and Toyohisa Kaneko. Automatic recognition of wild flowers. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 2, pages 507–510. IEEE, 2000.

[31] Chomtip Pornpanomchai, Ponrath Sakunreraratsame, Rosita Wongsasirinart, and Nuttakan Youngtavichavhart. Herb flower recognition system (hfrs). In *Electronics*

and Information Engineering (ICEIE), 2010 International Conference On, volume 1, pages V1–123. IEEE, 2010.

[32] Chomtip Pornpanomchai, Chawin Kuakiatngam Pitchayuk Supapattranon, and Nititat Siriwisesokul. Leaf and flower recognition system (e-botanist). *International Journal of Engineering and Technology (IACSIT). v3 i4*, pages 347–351, 2011.

[33] Siu-Yeung Cho and Peh-Ti Lim. A novel virus infection clustering for flower images identification. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 2, pages 1038–1041. IEEE, 2006.

[34] Andréa Britto Mattos, Ricardo Guimarães Herrmann, Kelly Kiyumi Shigeno, and Rogerio Schmidt Feris. Flower classification for a citizen science mobile app. In *Proceedings of International Conference on Multimedia Retrieval*, page 532. ACM, 2014.

[35] Milan Šulc and Jiří Matas. Image-based recognition of plants. Bachelor thesis, Czech Technical University in Prague, 2012.

[36] S Arivazhagan, R Newlin Shebiah, S Selva Nidhyanandhan, and L Ganesan. Fruit recognition using color and texture features. *Journal of Emerging Trends in Computing and Information Sciences*, 1(2):90–94, 2010.

[37] L Yang, J Dickinson, QMJ Wu, and S Lang. A fruit recognition method for automatic harvesting. In *Mechatronics and Machine Vision in Practice, 2007. M2VIP 2007. 14th International Conference on*, pages 152–157. IEEE, 2007.

[38] Tadhg Brosnan and Da-Wen Sun. Improving quality inspection of food products by computer vision—-a review. *Journal of Food Engineering*, 61(1):3–16, 2004.

[39] Dae Gwan Kim, Thomas F Burks, Jianwei Qin, and Duke M Bulanon. Classification of grapefruit peel diseases using color texture feature analysis. *International journal of Agricultural & Biological Engineering*, 2(3), 2009.

[40] Jagadeesh D Pujari, Rajesh Yakkundimath, and AS Byadgi. Reduced color and texture features based identification and classification of affected and normal fruits' images. 2013.

[41] Sergio Cubero, Nuria Aleixos, Enrique Moltó, Juan Gómez-Sanchis, and Jose Blasco. Advances in machine vision applications for automatic inspection and quality evaluation of fruits and vegetables. *Food and Bioprocess Technology*, 4(4):487–504, 2011.

[42] Robert M Haralick, Karthikeyan Shanmugam, and Its' Hak Dinstein. Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, (6):610–621, 1973.

[43] Joan S Weszka, Charles R Dyer, and Azriel Rosenfeld. A comparative study of texture measures for terrain classification. *Systems, Man and Cybernetics, IEEE Transactions on*, (4):269–285, 1976.

[44] Joan S Weszka and Azriel Rosenfeld. An application of texture analysis to materials inspection. *Pattern Recognition*, 8(4):195–200, 1976.

[45] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):2037–2041, 2006.

[46] Guoying Zhao and Matti Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):915–928, 2007.

[47] Gaurav Sharma, Sibt ul Hussain, and Frédéric Jurie. Local higher-order statistics (lhs) for texture categorization and facial analysis. In *Computer Vision–ECCV 2012*, pages 1–12. Springer, 2012.

[48] Jianguo Zhang, Marcin Marszałek, Svetlana Lazebnik, and Cordelia Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2):213–238, 2007.

[49] Laura Walker Renninger and Jitendra Malik. When is scene identification just texture recognition? *Vision research*, 44(19):2301–2311, 2004.

[50] Jianguo Zhang and Tieniu Tan. Brief review of invariant texture analysis methods. *Pattern recognition*, 35(3):735–747, 2002.

[51] Majid Mirmehdi, Xianghua Xie, and Jasjit Suri. *Handbook of texture analysis*. Imperial College Press, 2009.

[52] Chi-hau Chen, Louis-François Pau, and Patrick Shen-pei Wang. *Handbook of pattern recognition and computer vision*. World Scientific, 2010.

[53] Laurent Sifre and Stephane Mallat. Rotation, scaling and deformation invariant scattering for texture discrimination. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 1233–1240. IEEE, 2013.

[54] Peihua Li, Qilong Wang, and Lei Zhang. A novel earth mover's distance methodology for image matching with gaussian mixture models. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2013.

[55] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. *arXiv preprint arXiv:1311.3618*, 2013.

[56] 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.*, volume 1, pages 582–585 vol.1, 1994.

[57] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996.

[58] Timo Ojala, Kimmo Valkealahti, Erkki Oja, and Matti Pietikäinen. Texture discrimination with multidimensional distributions of signed gray-level differences. *Pattern Recognition*, 34(3):727–739, 2001.

[59] Topi Mäenpää and Matti Pietikäinen. Multi-scale binary patterns for texture analysis. In *Image Analysis*, pages 885–892. Springer, 2003.

[60] Guoying Zhao, Timo Ahonen, Jiri Matas, and Matti Pietikainen. Rotation-invariant image and video description with local binary pattern features. *Image Processing, IEEE Transactions on*, 21(4):1465–1477, 2012.

[61] Zhenhua Guo, Lei Zhang, and David Zhang. Rotation invariant texture classification using lbp variance (lbpv) with global matching. *Pattern recognition*, 43(3):706–719, 2010.

[62] Zhenhua Guo and David Zhang. A completed modeling of local binary pattern operator for texture classification. *Image Processing, IEEE Transactions on*, 19(6):1657–1663, 2010.

[63] Matti Pietikäinen, Timo Ojala, and Zelin Xu. Rotation-invariant texture classification using feature distributions. *Pattern Recognition*, 33(1):43–52, 2000.

[64] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *PAMI*, 34(3), 2011.

[65] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *arXiv preprint arXiv:1209.1873*, 2012.

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

[67] John Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3), 1999.

[68] Hsuan-Tien Lin, Chih-Jen Lin, and Ruby C Weng. A note on platt's probabilistic outputs for support vector machines. *Machine learning*, 68(3), 2007.

[69] Kimmo Valkealahti and Erkki Oja. Reduced multidimensional co-occurrence histograms in texture classification. *PAMI*, 20(1):90–94, 1998.

[70] Phil Brodatz. *Textures: a photographic album for artists and designers*, volume 66. Dover New York, 1966.

[71] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. A sparse texture representation using local affine regions. *PAMI*, 27(8):1265–1278, 2005.

[72] Eric Hayman, Barbara Caputo, Mario Fritz, and Jan-Olof Eklundh. On the significance of real-world conditions for material classification. In *Computer Vision-ECCV 2004*, pages 253–266. Springer, 2004.

[73] Mario Fritz, Eric Hayman, Barbara Caputo, and Jan-Olof Eklundh. The kth-tips database, 2004.

[74] Barbara Caputo, Eric Hayman, and P Mallikarjuna. Class-specific material categorisation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1597–1604. IEEE, 2005.

[75] P Mallikarjuna, M Fritz, AT Targhi, E Hayman, B Caputo, and JO Eklundh. The kth-tips and kth-tips2 databases. `http://www.nada.kth.se/cvap/databases/kth-tips`, 2006.

[76] Kristin J Dana, Bram Van Ginneken, Shree K Nayar, and Jan J Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics (TOG)*, 18(1):1–34, 1999.

[77] Yong Xu, Hui Ji, and Cornelia Fermüller. Viewpoint invariant texture description using fractal analysis. *International Journal of Computer Vision*, 83(1):85–100, 2009.

[78] Josef Kittler, Mohamad Hatef, Robert PW Duin, and Jiri Matas. On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):226–239, 1998.

Appendices

A Contents of the enclosed DVD

Directory	Content description
src	The MATLAB scripts for the proposed FSRIT method
lib	The VLFEAT open source library with added CLBP implementation
thesis	This thesis in PDF

Table 8: DVD contents description

B Publicly available datasets

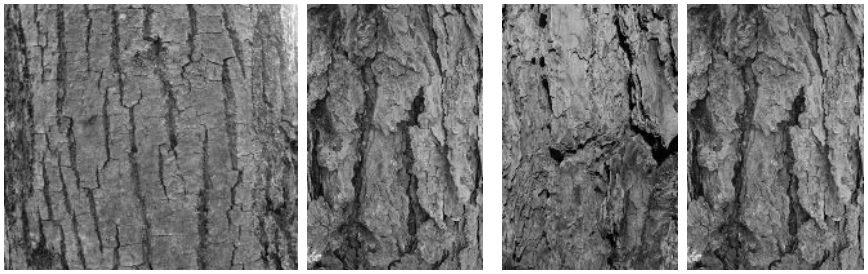
Name	URL
Flavia	`http://flavia.sourceforge.net`
Foliage	`http://rnd.akakom.ac.id/foliage`
Swedish	`http://www.isy.liu.se/cvl/ImageDB/public/blad`
Middle European Woods (MEW 2012)	`http://zoi.utia.cas.cz/node/662`

Table 9: Leaf datasets

Name	URL
UIUCTex	`http://www-cvr.ai.uiuc.edu/ponce_grp/data`
KTH-TIPS and KTH-TIPS2	`http://www.nada.kth.se/cvap/databases/kth-tips`
CUReT + images of selected regions	`http://www1.cs.columbia.edu/CAVE/software/curet` `http://www.robots.ox.ac.uk/~vgg/research/texclass`
UMD	`http://www.cfar.umd.edu/~fer/website-texture/texture.htm`

Table 10: Texture datasets

C Visualisation of misclassified samples

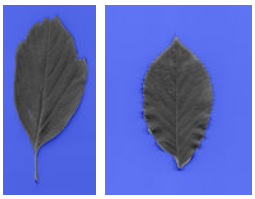


(a) Ash misclassified as Scots pine (b) Black pine misclassified as Scots pine

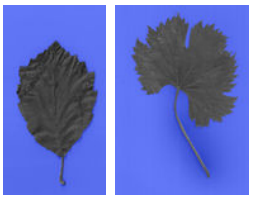
(c) Fir misclassified as Larch (d) Larch misclassified as Scots pine (e) Larch misclassified as Scots pine

(f) Scots pine misclassified as Larch (g) Swiss stone pine misclassified as Scots pine

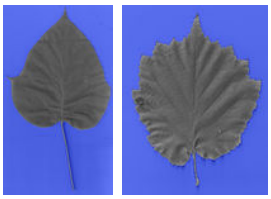
Figure 21: Examples of misclassification on the AFF bark dataset
(left: query image, right: example from wrongly assigned class)



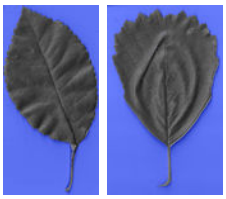
(a) Acer negundo misclassified as Sambucus nigra



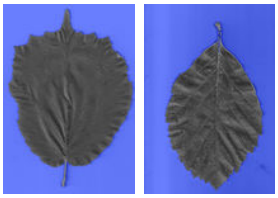
(b) Alnus incana misclassified as Vitis vinifera



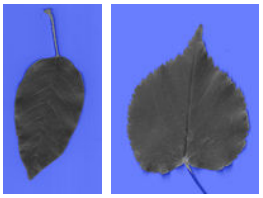
(c) Catalpa bignonioides misclassified as Corylus colurna



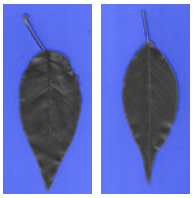
(d) Chaenomeles japonica misclassified as Alnus glutinosa



(e) Corylus avellana misclassified as Sorbus aria

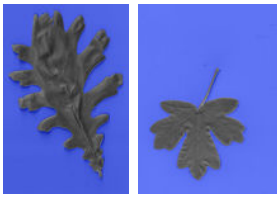


(f) Maclura pomifera FEMALE misclassified as Tilia cordata



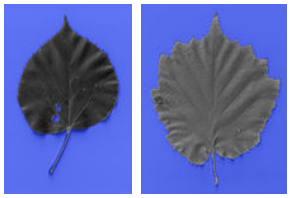
(g) Maclura pomifera FEMALE misclassified as Maclura pomifera MALE

(h) Mahonia aquifolium misclassified as Prunus mahaleb

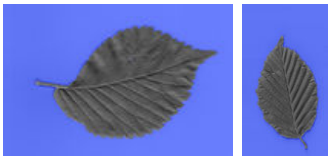


(i) Quercus frainetto misclassified as Acer campestre

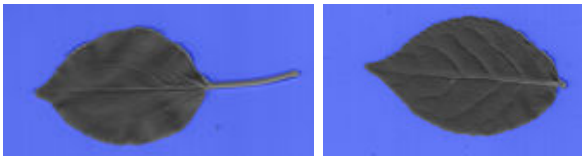
(j) Quercus rubra misclassified as Quercus robur



(k) Tilia cordata misclassified as Corylus colurna



(l) Ulmus minor misclassified as Carpinus betulus



(m) Rhamnus cathartica misclassified as Euonymus verrucosa

Figure 22: Examples of misclassification on the MEW leaf dataset
(left: query image, right: example from wrongly assigned class)