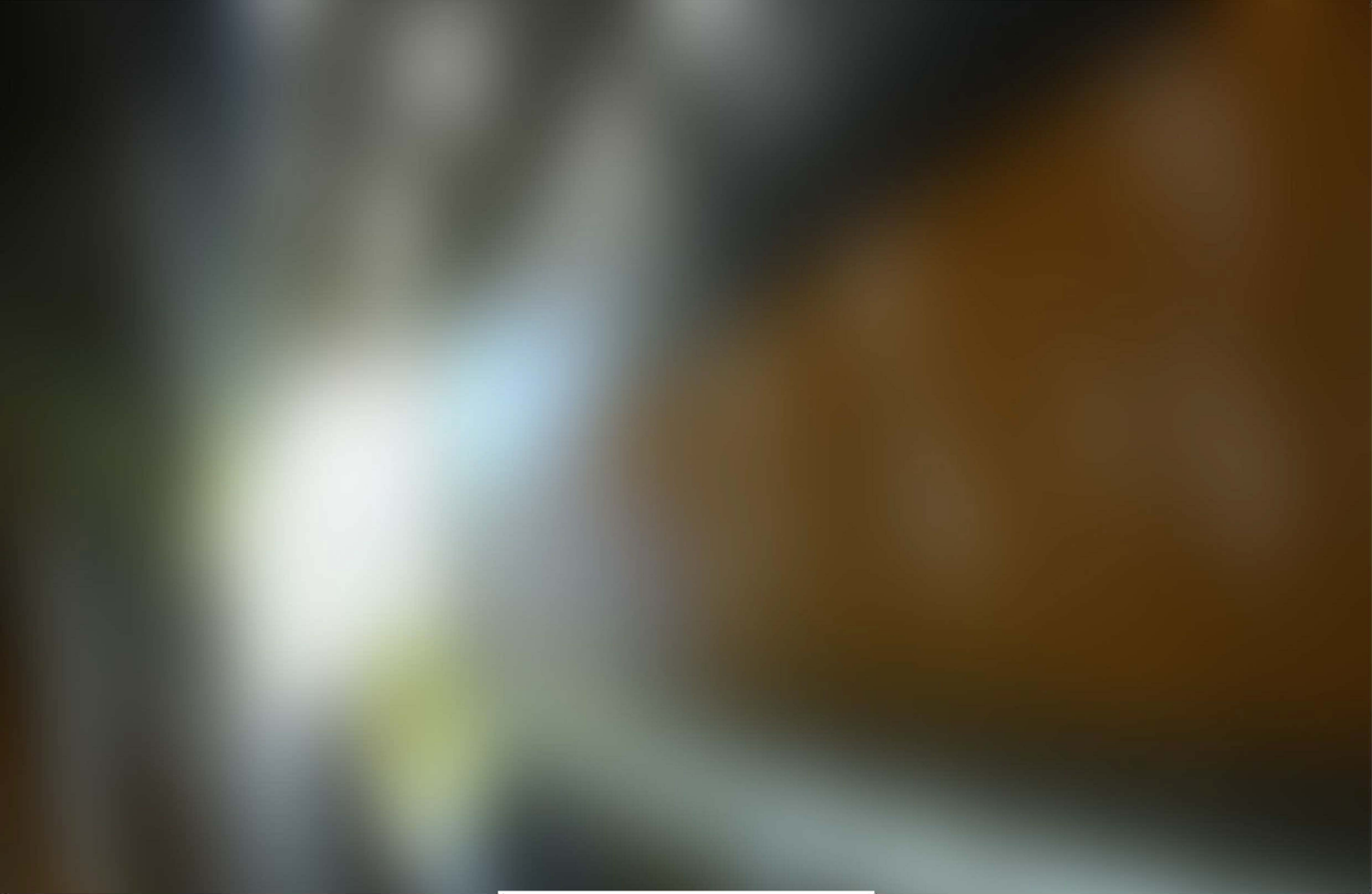
Kernel-based learning on hierarchical image representations : applications to remote sensing data classification

Yanwei Cui



UNIVERSITE BRETAGNE LOIRE

THESE / UNIVERSITE DE BRETAGNE-SUD

sous le sceau de l'Université Bretagne Loire

Pour obtenir le grade de :
DOCTEUR DE L'UNIVERSITE DE BRETAGNE-SUD

Mention : STIC
École Doctorale SICMA

présentée par

Yanwei CUI

IRISA Institut de Recherche en Informatique et Systèmes Aléatoires

Kernel-based learning on hierarchical image representations: applications to remote sensing data classification

Soutenance de thèse prévue le 04 Juillet 2017,
devant le jury composé de :

Luc Brun
Professeur, Ecole Nationale Supérieure d'Ingénieurs de Caen (ENSICAEN), France / Rapporteur

Lorenzo Bruzzone
Professeur, University of Trento, Italie / Rapporteur

Philippe-Henri Gosselin
Professeur, Ecole nationale supérieure de l'électronique et de ses applications (ENSEA), France / Examinateur

Ewa Kijak
Maître de Conférences, Université de Rennes 1, France / Examinateur

Laetitia Chapel
Maître de Conférences, Université de Bretagne-Sud, France / Encadrant

Sébastien Lefèvre
Professeur, Université de Bretagne-Sud, France / Directeur de thèse

Acknowledgements

I have finished one of the most interesting and challenging experiences during my Ph.D. Here I would like to thank my two wonderful supervisors: Sébastien Lefèvre and Laetitia Chapel, for their companionship. Their guidances and advices helped me to achieve the goals in an efficient way and their critical opinions allow me to think more deeply and more carefully. Thanks also the colleagues in IRISA, especially Nicolas Courty, Bharath Bhushan Damodaran, for the insight discussions and their inspirations. A special thanks for Manoj Joseph Mathew in geology department at UBS for helping me to improve English communication skills and for the joyful moments.

I am very grateful for all the reviewers of this thesis. I appreciate their interests in my work as well as all of their insightful comments and suggestions.

I acknowledge the support of the French Agence Nationale de la Recherche (ANR) under Reference ANR-13-JS02-0005-01 (Asterix project) and the support of Région Bretagne and Conseil Général du Morbihan (ARIA doctoral project) for the financial support of the Ph.D. program.

A special thanks for Anne Puissant from LIVE UMR CNRS 7362 (University of Strasbourg) for the warm welcome and discussion during my stay at University of Strasbourg, also for providing the datasets for evaluating the Ph.D. works.

I also want to express my gratitude to my parents for their unconditional love and support, and my dear friend Zhiying Shen, for the understanding and encouragement that helped me go through the difficult moments.

Abstract:

Hierarchical image representations have been widely used in the image classification context. Such representations are capable of modeling the content of an image through a tree structure, where objects-of-interest (represented by the nodes of the tree) can be revealed at various scales, and where the topological relationship between objects (*e.g.* A is part of B, or B consists of A) can be easily captured thanks to the edges of the tree. However, for fully benefiting from this key information, dedicated machine learning methods that can directly learn on hierarchical representations and handle the induced structured data need to be developed. In this thesis, we investigate kernel-based strategies that make possible taking input data in tree-structured and capturing the topological patterns inside each structure through designing structured kernels. We apply the designed kernel to remote sensing image classification tasks, allowing the discovery of complex cross-scale patterns in hierarchical image representations.

We develop a structured kernel dedicated to unordered tree and path (sequence of nodes) structures equipped with numerical features, called Bag of Subpaths Kernel (BoSK). BoSK is an instance of a convolution kernel relying on subpath substructures, more precisely a bag of all paths and single nodes. It is formed by summing up kernels computed on all pairs of subpaths of the same length between two bags. The direct computation of BoSK can be done through an iterative scheme, yielding a quadratic complexity *w.r.t.* both structure size (number of nodes) and amount of data (training size). However, such complexity prevents BoSK to be used on real world large-scale problems, where the tree can have more than hundreds of nodes and the available training data can consist in more than ten thousands samples. Therefore, we propose a fast version of the algorithm, called Scalable BoSK (SBoSK for short), using Random Fourier Features to map the structured data in a randomized finite-dimensional Euclidean space, where inner product of the transformed feature vector approximates BoSK. It brings down the complexity from quadratic to linear *w.r.t.* structure size and amount of data, making the kernel compliant with the large-scale machine learning context.

Thanks to (S)BoSK, we can learn from cross-scale patterns in hierarchical image representations. (S)BoSK operates on paths, thus allowing modeling the context of a pixel (leaf of the hierarchical representation) through its ancestor regions at multiple scales. Such a model is used within pixel-based image classification. (S)BoSK also deals with trees, making the kernel able to capture the composition of an object (top of the hierarchical representation) and the topological relationships among its subparts. This strategy allows tile/sub-image classification. Further relying on (S)BoSK, we introduce a novel multi-source classification approach that performs classification directly from a hierarchical image representation built from two images of the same scene taken at different resolutions, possibly with different modalities. Evaluations on several publicly available datasets illustrate the superiority of (S)BoSK compared to state-of-the-art remote sensing classification methods in terms of classification accuracy, and experiments on a urban classification task show the effectiveness of the proposed multi-source classification approach.

Keywords: structured kernel; image classification; hierarchical representations; Random Fourier Features; kernel approximation; large-scale machine learning; remote sensing

Résumé:

La représentation d'image sous une forme hiérarchique a été largement utilisée dans un contexte de classification. Une telle représentation est capable de modéliser le contenu d'une image à travers une structure arborescente, où les objets d'intérêt (représentés par les nœuds de l'arbre) peuvent être appréhendés à différentes échelles et où la relation topologique entre les objets (par exemple "A fait partie de B", ou "B se compose de A") peut être facilement décrite grâce aux arêtes de l'arbre. Cependant, pour bénéficier pleinement de ces informations-clés, des méthodes d'apprentissage statistiques doivent être développées pour traiter directement les données structurées sous leur forme hiérarchique. Dans cette thèse, nous considérons les méthodes à noyaux qui permettent de prendre en entrée des données sous une forme structurée et de tenir compte des informations topologiques présentes dans chaque structure en concevant des noyaux structurés. Nous appliquons le noyau que nous avons développé aux tâches usuelles de classification des images de télédétection, permettant ainsi de découvrir des modèles complexes dans les représentations hiérarchiques des images.

Nous présentons un noyau structuré dédié aux structures telles que des arbres non ordonnés et des chemins (séquences de nœuds) équipés d'attributs numériques. Le noyau proposé, appelé Bag of Subpaths Kernel (BoSK), est une instance du noyau de convolution et s'appuie sur l'extraction de sous-structures de sous-chemins, plus précisement un sac de tous les chemins et des nœuds simples. Il est formé en sommant les noyaux calculés sur toutes les paires de sous-chemins de même longueur entre deux sacs. Le calcul direct de BoSK peut se faire selon un schéma itératif, amenant à une complexité quadratique par rapport à la taille de la structure (nombre de nœuds) et la quantité de données (taille de l'ensemble d'apprentissage). Cependant, une telle complexité ne permet pas d'utiliser BoSK pour résoudre des problèmes à grande échelle, où la structure peut contenir des centaines de nœuds et les données d'apprentissage disponibles peuvent comporter plus de dix milliers d'échantillons. Par conséquent, nous proposons également une version rapide de notre algorithme, appelé Scalable BoSK (SBoSK), qui s'appuie sur les Random Fourier Features pour projeter les données structurées dans un espace euclidien, où le produit scalaire du vecteur transformé est une approximation de BoSK. Cet algorithme bénéficie d'une complexité non plus quadratique mais linéaire par rapport aux tailles de la structure et de l'ensemble d'apprentissage, rendant ainsi le noyau adapté aux situations d'apprentissage à grande échelle.

Grâce à (S)BoSK, nous sommes en mesure d'effectuer un apprentissage à partir d'informations présentes à plusieurs échelles dans les représentations hiérarchiques d'image. (S)BoSK fonctionne sur des chemins, permettant ainsi de tenir compte du contexte d'un pixel (feuille de la représentation hiérarchique) par l'intermédiaire de ses régions ancêtres à plusieurs échelles. Un tel modèle est utilisé dans la classification des images au niveau pixel. (S)BoSK fonctionne également sur les arbres, ce qui le rend capable de modéliser la composition d'un objet (racine de la représentation hiérarchique) et les relations topologiques entre ses sous-parties. Cette stratégie permet la classification des tuiles ou parties d'image. En poussant plus loin l'utilisation de (S)BoSK, nous introduisons une nouvelle approche de classification multi-source qui effectue la classification directement à partir d'une représentation hiérarchique construite sur deux images de la même scène prises à différentes résolutions, éventuellement selon différentes capteurs. Les évaluations sur plusieurs jeux de données de télédétection disponibles dans la communauté illustrent la supériorité de (S)BoSK par rapport à l'état de l'art en termes de précision de classification, et les expériences menées sur une tâche de classification urbaine montrent la pertinence de l'approche de classification multi-source

proposée.

Mots clés: Noyau structuré; Classification d'image; Représentations hiérarchiques; Random Fourier Features; Approximation du noyau; Apprentissage à grande échelle; Télédétection

Contents

Chapter 1

Introduction

Contents

1.1 Context: classification of remotely-sensed images

1.1.1 Classification in remote sensing: objectives and challenges

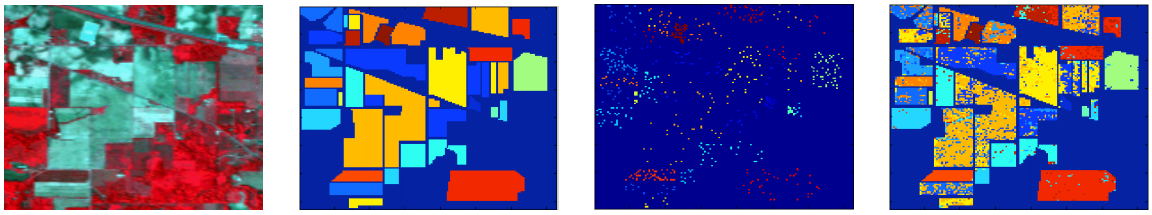
Remote sensing is considered as one of the most effective ways for Earth observation. It is generally defined as the technology that measures the surface of Earth from remote, where the acquisition of images can be obtained with some satellite or airborne sensors. Through remote sensing image analysis, we can achieve an accurate identification of materials or even complex objects on the surface of the Earth. Therefore, it provides valuable information for various applications, which include but are not limited to:

- precision agriculture — remote sensing images are used to identify different types of crops and monitor different changes of these crops;
- disaster management — affected areas can be quickly accessed with remote sensing images, providing the possibilities for a rapid damage assessment;
- Urban planning — urban development can be monitored through the remote sensing image archives acquired at different times, with applications like road map updating, or change detection in urban areas.

Although we can acquire a large amount of remote sensing archives every day, images without processing can hardly provide any useful information. One of the most important tasks is image classification, whose goal is to summarize the image into a predefined (according to some specific applications) list of classes, thus providing fundamental resources that can be easily reused in the next steps of a decision making process. Techniques able to automatically classify images have attracted the attention of researchers for several decades [62, 15, 78]. An example of remote sensing image classification is given in Fig. 1.1.

In the context of remote sensing image classification, several new challenges have emerged because of the recent development of remote sensing sensors:

- **High resolution.** The challenges raised by high resolution in digital images came from both spectral and spatial domains. In the spectral domain, the hyperspectral image sensors allow the acquisition of the signal in hundreds of spectral wavelengths for each image pixel, which can later provide high discrimination capabilities towards identification of different species or material. However, the resulting images have large correlated dimensions, which induce problems related to the high dimensionality of data especially in the case of limited availability of training samples [15]. Techniques able to effectively exploit the high dimensionality of the images still need to be investigated. Another challenge is related to high spatial resolution. The recent availability



(a) False color image (b) Ground truth (c) Training samples (d) Classification map

Figure 1.1: An example of pixel-wise remote sensing image classification. The false color composition of hyperspectral image Indian Pines is given in Fig. 1.1a, together with its associated ground truth (mostly unknown for real-world applications) for reference in Fig. 1.1b. After selecting a small number of pixels for training like in Fig. 1.1c, we obtain the classification map given in Fig. 1.1d.

of Very High Spatial Resolution (VHSR) images provides submetric resolution, allowing the exploitation of the fine details of the observed scene. Thanks to new sensors, additional information such as texture, shape of complex objects or even structure of the object composition can be better revealed. However, the way to take into account these information remains challenging [62].

- **Multi-modal.** We face nowadays a large number of sensors with different spatial resolutions. For instance, we can rely on high resolution sensors such as Quickbird with 0.6 m per pixel, Pleiades with 0.5 m, while low and medium resolution sensors are still in use, *e.g.* SPOT-4. Multiple remote sensing image sources are available for the same geographical region, and these sources can be fused to improve classification accuracy. In this context, data fusion techniques that can make better use of various source still need to be elaborated [78].

- **Large volume.** Another challenge is related to the big volume of remote sensing image archives raised by the latest generation of remote sensing sensors. For instance, the average volume of data acquired from a single satellite is about more than 500 GB every day [126]. These data make it possible to monitor Earth at a global scale. However, the efficient processing of such data remains largely unexploited [37, 118].

In this thesis, we address the above mentioned challenges. Readers are referred to [15, 29, 192] for other challenges faced in the field, such as limited availability of training samples, mixed pixels in low spatial resolution images, or domain adaptation. These challenges have not been addressed in this manuscript.

1.1.2 Principles of remote sensing classification

In the context of machine learning, supervised classification aims to find a "rule" based on available data together with their class labels, and use the constructed "rule" to assign new

data to one of the classes. The available data are called training data, which can be represented as $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$, where x_i is the data instance and y_i is the associated class label.

The process of remote sensing image classification is composed of data representation, feature extraction, classifier training and prediction. The final quality of the classification depends on the performance of each single step.

The first step is **data representation**, whose goal is to define what kind of element will be classified and how it will be described (*i.e.* with which features). In remote sensing, input data can take various forms: image pixels (leading to the so-called pixel-wise classification); regions, commonly used in the object-based image analysis paradigm [36]; or complex structures such as those obtained with hierarchical image representations [17].

Feature extraction step depends highly on data representation. For instance, in pixel-wise classification, a pixel is described by its spectral information encoded in a d-dimensional vector, while region-based classification uses more complex features such as size, shape of the region, or even texture information. In this thesis, we focus on hierarchical representations, where not only the features of regions that are used, but also the hierarchical relationships among the regions are encoded through complex structured data. We especially focus on tree-based structures *i.e.* tree and sequence of nodes.

Then we feed the data into a **classifier**. Various classifiers have been applied for remote sensing classification, which includes K-nearest neighbors [42], Support Vector Machines [139], Random Forests [11, 150], Neural Networks [10] and Deep Neural Networks [94]. SVM is a popular kernel-based method in the remote sensing community and has been investigated for the past 20 years, as its efficiency and capacities to provide high classification accuracies, especially when only limited training samples are available [139]. In addition, it is able to handle complex structured data.

The **outputs of classifier** (predictions) are generally used directly as the classification map. However, in real world applications, manually editing or automatized post-processing techniques might be applied for generating "smoother" classification maps [13, 183].

Following the standard classification scheme, different research topics are brought into the literature for improving the classification accuracies. The following section gives a brief review of main trends for remote sensing image classification.

1.1.3 Main trends for remote sensing image classification

In this section, we give a brief review of several main trends in remote sensing image classification that are highly related to the problems addressed in this thesis.

Pixel-wise classification using contextual information

In most of conventional approaches in pixel-wise classification, each pixel is treated independently. However it is not appropriate, as neighboring pixels are highly correlated and are more likely belong to the same class. This is particular true for very high spatial resolution remote sensing imagery. Due to the recent advances in sensor technology that enable submetric resolution, the spectral variance inside same classes has been highly increased. Conventional techniques that only use spectral information often produce high ratio of classification errors [208]

In the literature, integrating contextual information is considered as one of the key solutions to the aforementioned issue. As pointed out in several recent survey papers [62, 76], including spatial contextual information can reduce the labeling uncertainty by exploiting additional discriminant information *e.g.* the shape and size of different structures formed by neighboring pixels. Moreover, with contextual information, more spatially "smoother" classification maps can be produced.

Contextual information associated with each pixel can be modeled through the neighboring pixels. For instance, in [61], the pixel contextual information is represented as the median of spectral features of the region the pixel belongs to. Another representative example are attribute profiles [76]. They integrate spatial contextual information using morphological filtering around pixels with different attributes, *e.g.* size, spectral standard deviation, and produce a high dimension feature vector that encodes the changes of each pixel under different filterings.

Due to the importance of taking into account contextual information, various pixel-wise classification methods have been adapted in the different steps of the classification scheme: spatial-spectral features extraction [76, 188], classifier that incorporates pixel spatial neighborhood [204, 165], and post-processing of the final classification maps [183, 197]. Exploiting efficiently the contextual information is one hot research topic for pixel-wise remote sensing image classification.

Content-based tile classification with spatial decomposition patterns

In the remote sensing community, a large number of applications are related to land-use classification of high resolution images. Current approaches concentrate in splitting the observed scene taken from large Earth surface into small tiles, and process each tile independently [95]. Therefore, the data instances are represented as single images (or tiles), where the objective is to assign each image one of the predefined labels.

In these applications, labels are commonly associated with some semantics *e.g.* in the land-use classification context, where the observed scene contains various types of complex

objects and the spatial organization of these objects reveals semantic information on the surface. For instance, sparse, medium, dense residential areas, mobile home are defined as different classes in one standard land-use remote sensing dataset [213]. All these classes share common objects such as roads, trees and buildings, but the difference in the density and spatial distribution of these objects might serve as discriminative information in different classes. Therefore, it is still challenging to model such complex patterns for various land-use classification applications.

One of the key solutions is to exploit the spatial arrangement of objects and structural patterns present within the image. At a local scale, the structural patterns can be interpreted as the textural information within a small patch or region and can be captured by local descriptors such as SIFT [123] or Local Pattern Spectra [23]. Structural patterns can also be described at a global scale, revealing spatial relationships among complex objects/regions.

Following this direction, a large number of works rely on patch/region-based local feature extraction and quantification into an orderless organized histogram [95], commonly known as "bag-of-words", one of the most successful approaches in the computer vision community [33]. Taking into account the spatial relationships among patches/subregions has been proved to be effective in many applications [213, 214]. More recently, some techniques relying on hierarchical spatial decomposition of the observed scene into several subregions demonstrate promising performances in land-use classification [35].

Multimodal classification of remote sensing images

Multimodal classification is becoming an important topic in the remote sensing community, thanks to the recent technologies that make the multiple and heterogeneous image sources available for the same geographical Earth surface area. The research in remote sensing data fusion becomes very active in the recent years. Indeed, data fusion contest is held annually by IEEE Geoscience and Remote Sensing Society in order to encourage the development of new methods [50, 190].

Facing the challenges brought by the large amount of images coming with different resolutions (spatial, spectral, temporal) and from heterogeneous sources (SAR, LiDAR), data fusion techniques have demonstrated the interest of exploiting the complementary information of the observed scene carried by the different modalities. Techniques able to fuse data from multiple sources and multiple resolutions have been proved to be effective for improving classification accuracy [218]. As each sensor provides some unique spatial details from the observed scene, exploring and combining these information becomes crucial. In order to tackle the challenges raised by image classification with multiple sources and multiple resolutions, various methods have been proposed in the literature. They actually occur in different steps of the classification process, which includes feature extraction [51], kernel

combination [194, 28] or fusion inside the classifier [174, 135], or even merging the outputs of each classifier [60].

Large-scale remote sensing image classification

Recent technologies make possible the acquisition of massive amounts of remote sensing images. Indeed, it has been reported that the volume of data acquired only in one data center (among many) could sum up to about 2 TB per day [126]. In the near future, the continuously increasing amount of data and variety of applications relying on remote sensing classification will bring new challenges and require more adaptive machine learning methods. From a research point of view, the availability of large-scale remote sensing data will certainly modify the development of remote sensing image classification algorithms. For example, the SpaceNet Challenge [1] includes more than 60 millions of labeled high-resolution images available for training. Such a publicly available large-scale benchmark requires the majority of current state-of-the-art methods adapting their scalability in order to be assessed in large-scale conditions. Techniques able to scale up machine learning methods have gained increasing attention recently in various domains, including image classification [4]. Recently, an increasing number of research papers concentrate on large-scale remote sensing image classification [127, 130, 134].

1.2 Hierarchical image representations and applications

1.2.1 Motivation for hierarchical representations

Different image representations exist in order to adapt to various applications. As the initial step of remote sensing image classification, data representation has a direct impact on the next steps. For such a reason, we can observe an evolution in the representation adopted, ranging from conventional pixel-based representation [208] to, more recently, hierarchical representation [17].

In the early stage of remote sensing image classification, pixel-based representation has been largely used. As the pixel represents the basic unit in an image, it is natural to represent an image as a set of pixels. However, in such a representation, each pixel is considered as an individual data instance and the dependence between neighboring pixel can not be revealed. Indeed, ignoring spatial relationship often produces higher ratio of classification errors [183, 197, 208].

In order to take into account the spatial information and consider the dependence of neighboring pixels, region-based representation has been proposed. A region is considered

[1] more details at http://explore.digitalglobe.com/spacenet

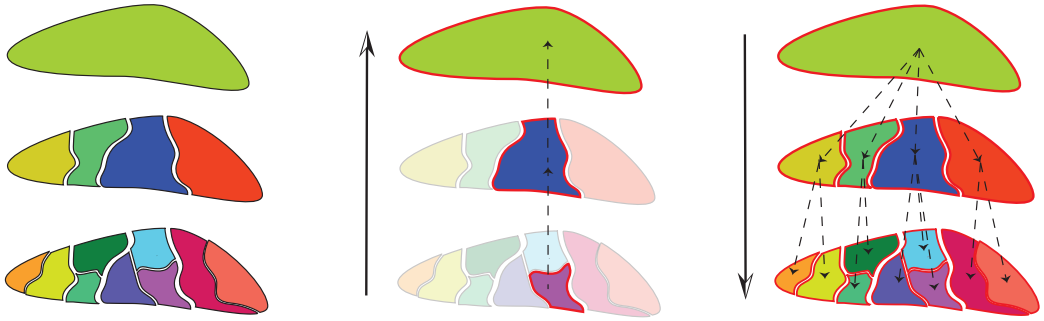


Figure 1.2: An illustration of hierarchy of image objects (left), revealing the spatial context of objects (middle), and objects spatial decomposition (right).

as a group of neighboring pixels that share some similar characteristics. It can be obtained by segmentation algorithms that partition image into different regions upon certain similarity criterion. In many applications, such as the GEOBIA (GEOgraphic Based Image Analysis) framework, image regions are considered as the processing units, called objects, and various characteristic can be extracted, *e.g.* shape, size, texture. However, the quality of the region might have a significant impact of underlying results. One challenge is related to the scale issue: various objects-of-interest might appear at different scales and segmentation scale parameters are difficult to be determined in advance without prior knowledge about the next analyses. For example, an analysis of individual buildings might require smaller region size than urban residential area. For this reason, common solutions use only empirical parameters, which often result in sub-optimal segmentations and degrade the classification accuracy.

Many researchers have dealt with hierarchical representations of digital images. Such representations highlight objects-of-interest at different scales, where the topological relationship between objects (*e.g.* A is part of B, or B consists of A) can be easily modeled. However, fully exploiting the key concepts of hierarchical representation, including incorporating contextual information through multiscale analysis, modeling the complex topological information revealed from hierarchical relationships among objects, is still considered as an open challenge [17].

In this work, we rely on hierarchical representations of images. More precisely, we aim at fully exploiting the key advantages of such representations, and especially including them in the conventional classification scheme in order to improve the quality of automated land cover/land use mapping results. More specifically, we concentrate on incorporating the different types of topological information across the scales from a hierarchical representation: contextual information and object spatial decomposition information, as illustrated in Fig. 1.2.

1.2.2 Construction of hierarchical representation

In this section, we recall the principles of hierarchical representations and the most popular algorithms to build them from digital images. We give a special attention to Hseg multiscale segmentation algorithm, one successful implementation of open-source segmentation tool for remote sensing image processing [185] [2], as it is used as the hierarchical representation generation tool in this thesis.

Hierarchical representations can be categorized into two classes[22]: inclusion trees and partition trees.

- **Inclusion trees** aim at representing bright and/or dark structures of the image. Leaves in inclusion trees are often image extrema, *e.g.* the minimum intensity value in the image, and inner nodes are formed by region growing from the leaves until the root which covers the whole image. In general, any cut of an inclusion tree does not form a complete partition of the underlying image. Typical examples are Max- and Min- tree [142], Tree of Shapes [74]. Both have been successfully used in remote sensing [30, 49].

- **Partitioning trees**, on the other side, are initialized from an image partition. Then they rely on iterative merges of small regions at finer scale into larger regions at coarser levels. Among typical examples, we can cite Binary Partition Trees (BPT) [164], α-tree [148], ω-tree [173]. Partitioning trees are commonly used in object-based remote sensing image representation, as the nodes in the tree correspond in general to the objects-of-interest.

Hseg can generate partitioning trees of arbitrary form according to some user definitions *e.g.* number of regions for each level. It relies on a BPT construction and outputs multiscale segmentation maps at various levels through thresholding. It starts with an initial partition to form leaves, *e.g.* pixels, flat zones, or pre-segmented regions. The algorithm then iteratively computes the similarity between all pairs of spatially adjacent nodes, and merges the two most similar ones until whole image being a single region. The computation of similarity is related to two key concepts: region model and merging criterion. The region model measures the characteristics of each region, while the merging criterion is the value of the difference or dissimilarity between the region models.

In Hseg, various region models based on spectral information have been proposed, allowing Hseg to handle multispectral or even hyperspectral images in a native way. One of the most basic region models is based on averaging the spectral information of pixels that compose the region:

[2] an open source software of NASA, which can be downloaded at https://opensource.gsfc.nasa.gov/projects/HSEG/

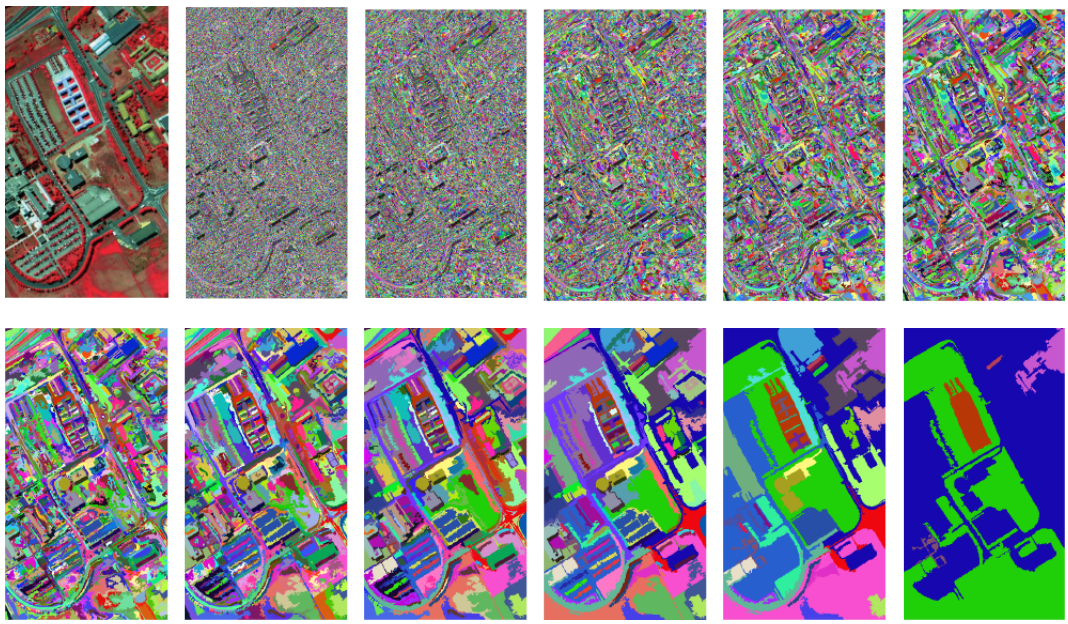


Figure 1.3: An illustration of multiscale segmentation on hyperspectral image captured at Pavia University. Original false color image is shown at beginning (top left), followed by 11 different segmentation maps generated using Hseg with dissimilarity criterion set respectively as $\alpha = [0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512]$.

$$\overline{R^b} = \frac{1}{Area(R)} \sum_{Pixel_i \in R} Pixel_i^b \,, \tag{1.1}$$

where R^b is the spectral band b of region R and its average $\overline{R^b}$, $Area(R)$ is the size of the region and $Pixel_i$ is one pixel in the region R with its spectral information of band b written as $Pixel_i^b$.

As far as the merging criterion is concerned, "Band Sum Mean Squared Error (MSE)" is used for evaluating the dissimilarity of one region versus another. It is defined as:

$$MSE(R_i, R_j) = \frac{Area(R_i)Area(R_j)}{Area(R_i) + Area(R_j)} \sum_{b=1}^{B} (\overline{R_i^b} - \overline{R_j^b})^2 \,. \tag{1.2}$$

Such a dissimilarity measures the distance between two regions, and is used to determine the merging order at each iteration step. Relying on the predefined dissimilarity criterion, Hseg can output multiscale segmentation maps, forming a partitioning tree in arbitrary form. Fig. 1.3 shows an example on multiscale segmentation maps using Hseg for hyperspectral remote sensing image, where the dissimilarity criterion $\alpha = \sqrt{MSE(R_i, R_j)}$ is set as $\alpha = [0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512]$.

1.2.3 Applications in remote sensing

Spatial-spectral classification

Spatial-spectral classification, which incorporates spatial information to improve the classification results, has became one of the most popular approaches for remote sensing classification [62, 76, 157, 25], especially when the image spatial resolution is high. In these approaches, spatial information is extracted at the image region level rather than at the conventional pixel level.

Therefore, one of the key application of hierarchical representation is the modeling of the context of a pixel. Through the hierarchy, the context models the evolution of a pixel and describes it at different scales [55, 25]. One of the most popular examples is attribute profiles, relying on Max- and Min- tree [48], or Tree of Shapes [30].

The Attribute Profile built on the image I can be written as:

$$AP(I) = \{\phi^{\lambda_L}(I), \phi^{\lambda_{L-1}}(I), \ldots, \phi^{\lambda_1}(I), I, \gamma^{\lambda_1}(I), \ldots, \gamma^{\lambda_{L-1}}(I), \gamma^{\lambda_L}(I)\}\ , \tag{1.3}$$

where the ϕ and γ stands for the thickening and thinning operator respectively, λ_L is the scale parameter at the level L. In such a setting, each pixel in the image can be represented as its corresponding values in $AP(I)$, a stacked vector of dimension $2L + 1$.

In addition to attribute profiles, other techniques have also been proposed to model contextual information with hierarchical representations. In [25], the authors state that given a hierarchical image representation, it is possible to exploit the relationships between pixels and regions at different levels to extract an effective set of features that describes each pixel and its adaptive context at each level. In [113], a similar multiscale context feature extraction strategy is proposed, but with a different hierarchical image representation called α-tree.

All the previous methods discussed so far share similar strategy when using the multiscale features. Once extracted, these features are concatenated into a long raw (*i.e.* flat, unstructured) vector, on which is applied a conventional vector-based machine learning technique (*e.g.* SVM with the Gaussian kernel). Such stacked vectors are usually sets of highly dimensional features. Consequently, as mentioned in [76], they should be properly handled in order to make full exploitation of the discriminative information, and avoid issues raised by very large dimensionality (Hughes phenomenon) and high redundancy [152].

Spatial pyramid matching model

The Spatial Pyramid Matching (SPM) model [109, 209] is the most common strategy to consider the object spatial decomposition based on a hierarchical representation. The idea is to segment the image in 4 regions at successive scales (through a quad-tree representation), and

to concatenate all the region features into a long vector. With such a hierarchical representation, objects and their subparts can be revealed in various scales, and the relative spatial arrangement of the objects and the object decomposition information can be also modeled. These features play a significant role in classification of complex patterns [214].

However, the matching strategy of SPM limits its application to quad-tree representations, preventing it to benefit from advanced multiscale segmentation techniques that lead to meaningful but mostly irregular hierarchical representation. In addition, SPM only allows matching image regions at the same spatial position. Therefore, applying SPM to remote sensing image classification is not optimal since SPM hardly adapts to images with no predefined location or orientation [221, 35, 213].

GEOBIA

GEOgraphic-Object-Based Image Analysis (GEOBIA) framework has gained increasing interest and is today a paradigm for remote sensing image processing [16, 17] beyond conventional pixel-based analysis. In this framework, hierarchical image representation is the key concept — meaningful objects are obtained from multiscale image segmentation, and spatial relationships among objects are encoded in a tree structure.

The main trends of GEOBIA have shifted from choosing the correct scale for analyzing objects [56], recognizing the different changes occurring at different scale for different type of objects [83], and also understanding the mutual relations between image objects [17]. Within the hierarchical representation, each object is characterized not only by segment-related characteristics, which are its spectral, shape or texture features, but also with the topological information (*e.g.* A is part of B, or B consists of A). In addition, semantic relationships between objects at different levels revealed from hierarchical representation allow effective multiscales analysis. For instance, building and tree species at finer level can form residential blocks at intermediary level, and groups of residential blocks can form urban area at coarse scale.

The first commercial object-based image analysis software is eCognition and triggered the major trend of object-based remote sensing image analysis. Most applications have employed this software with a rule set based classification framework. In [36], hierarchical rule-sets have been designed to incorporate the expert knowledge, *e.g.* thresholding the NDVI information is used for distinguish vegetation and non-vegetation, and height information, ratio of width and length of object, are further included to divide the non-vegetation into building, vacant land, and road. A similar rule-set based scheme can be found in [170] for mapping gully extraction using high spatial resolution imagery. In [121], the authors propose to rely on the object features and the spatial relations between objects to extract roads and moving vehicles from remote sensing imagery. Object features such as size, shape can

differentiate road and vehicle, and spatial relations between objects, *i.e.* moving vehicles are surrounded by a road, help to refine classification results and extract particularly moving vehicles.

The knowledge-based subjective rule-set strategy commonly used in GEOBIA is highly relying on human involvement and interpretation, which makes it difficult to adapt to new locations and datasets, and makes the processing of data in large remote sensing archives practically impossible. Advanced machine learning techniques are thus required for learning automatically properties of the objects and the relationships among them.

1.3 Kernel-based machine learning

1.3.1 Kernel definition

Kernels are popular in machine learning, thanks to their capability of capturing non-linear patterns in the data. They have been introduced to map data in a new feature space, where it becomes possible to separate linearly the transformed data.

Following the kernel definition in [166], let us define a kernel as a function $k(\cdot)$ that for $\boldsymbol{x}, \boldsymbol{x'} \in \mathcal{X}$ satisfies:

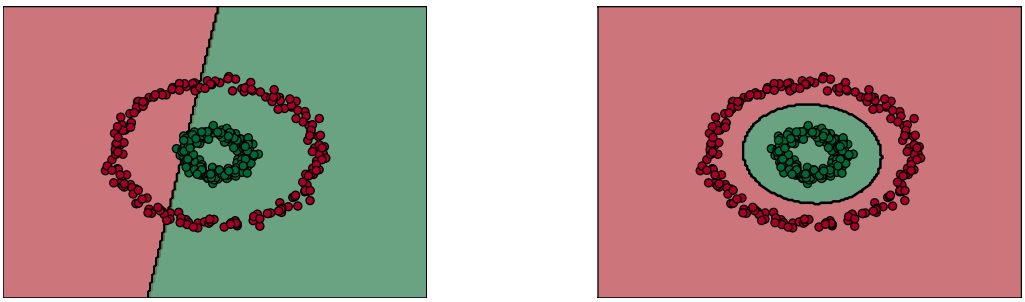
$$\begin{aligned} &k(\boldsymbol{x}, \boldsymbol{x'}) = \langle\, \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) \,\rangle_{\mathcal{H}} \,, \\ &\phi : \boldsymbol{x} \in \mathcal{X} \longmapsto \phi(\boldsymbol{x}) \in \mathcal{F} \,, \end{aligned} \tag{1.4}$$

where ϕ is a mapping from the original input space $\mathcal{X}$ to a feature space $\mathcal{F}$, and the inner product of two mapping $\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) \rangle_{\mathcal{H}}$ can be directly computed using kernel function on their original space $k(\boldsymbol{x}, \boldsymbol{x'})$.

A well known example is the polynomial kernel. Let $\boldsymbol{x} = [x_1, x_2] \in \mathbb{R}^2$ a two dimensional vector, and the mapping function is defined as $\phi : \boldsymbol{x} \longmapsto \phi(\boldsymbol{x}) = [x_1^2, x_2^2, \sqrt{2}x_1x_2] \in \mathbb{R}^3$. Fig. 1.5 shows an example of such a mapping, non-linearly separable data in original space $\mathbb{R}^2$ can become linearly separable in the feature space $\mathbb{R}^3$.

Further, we can compute the inner product between the projections of data without explicitly evaluating them in the feature space. Following the previous example, we have

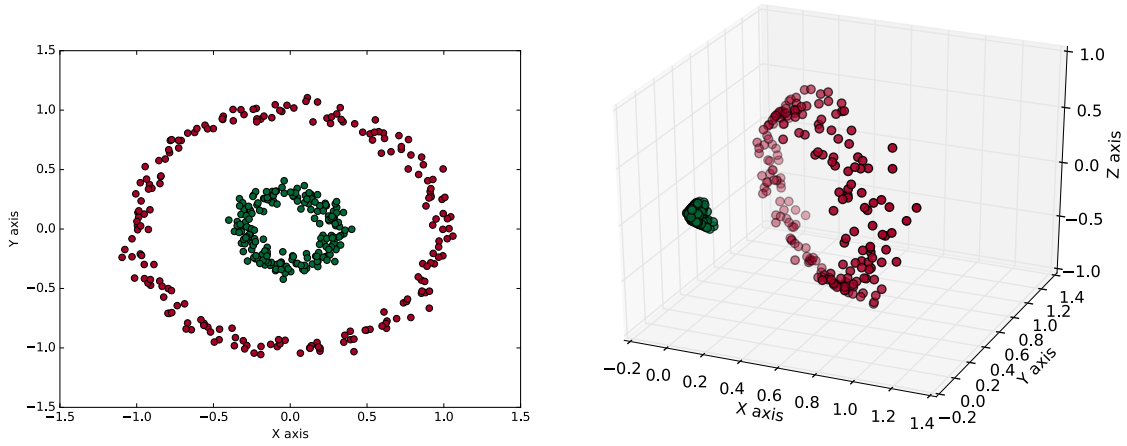
$$\begin{aligned} k(\boldsymbol{x}, \boldsymbol{x'}) &= < \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) > \\ &= [x_1^2, x_2^2, \sqrt{2}x_1x_2]^T \, [x'^2_1, x'^2_2, \sqrt{2}x'_1x'_2] \\ &= x_1^2x'^2_1 + x_2^2x'^2_2 + 2x_1x_2x'_1x'_2 \\ &= (x_1x'_1 + x_2x'_2)^2 = (\boldsymbol{x}^T\boldsymbol{x'})^2 \end{aligned} \tag{1.5}$$



(a) Decision boundary with linear SVM

(b) Decision boundary with kernel SVM

Figure 1.4: Illustration of non linear separable case with linear SVM and kernel SVM.



(a) Data in the input space $\boldsymbol{x} \in \mathbb{R}^2$

(b) Data in the feature space $\phi(\boldsymbol{x}) \in \mathbb{R}^3$

Figure 1.5: Illustration of data in the original input space and feature space.

where the kernel function $k(\boldsymbol{x}, \boldsymbol{x'}) = (\boldsymbol{x}^T \boldsymbol{x'})^2$ can be computed directly in the original data input space. This is known as the "kernel trick" and it is very helpful since the high dimensional feature space does not need to be projected firstly. In other cases, $\phi(\cdot)$ might be unknown, such as with the Gaussian kernel.

1.3.2 A kernel method example: Support Vector Machine (SVM)

We briefly introduce SVM in this section, interested readers are referred to [67] for a detailed description.

Linear SVM

Let us assume a binary classification problem that consists of N pairs $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2)..., (\boldsymbol{x}_N, y_N)$, with $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in= \{-1, +1\}$. The decision function

of SVM is defined as $f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{\beta} + \beta_0$ such that if $f(\boldsymbol{x}_i) < 0$, $y_i = -1$, $y_i = 1$ otherwise. In short, we have $y_i f(\boldsymbol{x}_i) \geq 0$, where $\boldsymbol{\beta}, \beta_0$ are the parameters of the model and need to be learned.

In order to learn the parameters $\boldsymbol{\beta}, \beta_0$, SVM minimizes the following objective function

$$\| \boldsymbol{\beta} \|^2 + C \sum_{i=1}^{N} \max(0, 1 - y_i f(\boldsymbol{x}_i)) \tag{1.6}$$

where $\| \boldsymbol{\beta} \|^2$ is the L_2 regularization on the parameters of the model, and the function $\max(0, 1 - y_i f(\boldsymbol{x}_i))$ is the hinge loss function. C is a hyper-parameter that controls the balance between the loss (how well the model fits on training data) and the regularization (the complexity of the model).

The geometric interpretation of SVM can be seen as the margin maximization problem, where the decision boundary is defined as a hyperplane $\boldsymbol{x}^T\boldsymbol{\beta} + \beta_0 = 0$, and the distance from the decision surface to the closest data point is called margin and is defined as $\frac{2}{\|\boldsymbol{\beta}\|}$, as shown in Fig. 1.6. While maximizing the margin, SVM can be rephrased into a constrained optimization problem:

$$\begin{aligned} &\min_{\boldsymbol{\beta}} \| \boldsymbol{\beta} \| \\ &s.t.\ y_i(\boldsymbol{x}_i^T\boldsymbol{\beta} + \beta_0) \geq 1 - \xi_i \\ &\xi_i \geq 0; \sum \xi_i \leq Constant \end{aligned} \tag{1.7}$$

where ξ_i is called slack variable that allows the violation of the margin at the non-negative cost of ξ_i. The sum of all costs ξ_i is bounded by a constant, which limits the total number of predictions on the wrong side of its margin.

One might notice that the optimization problem of Eq. (1.6) and Eq. (1.7) leads to the same solution, and they can be further rewritten in an equivalent form, often called primal form:

$$\begin{aligned} &\min_{\boldsymbol{\beta}} \ \frac{1}{2} \| \boldsymbol{\beta} \|^2 + C \sum_{i=1}^{N} \xi_i \\ &\text{s.t. } y_i(\boldsymbol{x}_i^T\boldsymbol{\beta} + \beta_0) \geq 1 - \xi_i;\ \ \xi_i \geq 0 \end{aligned} \tag{1.8}$$

The above constrained optimization problem can be rewritten in its dual form (see [67] for detailed derivation):

$$\begin{aligned} &\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} \alpha_i \alpha_{i'} y_i y_{i'} x_i^T x_{i'} \\ &\text{s.t. } 0 \leq \alpha_i \leq C; \ \sum_{i=1}^{N} \alpha_i y_i = 0 \end{aligned} \tag{1.9}$$

The decision function derived from dual formulation can be written as:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i (x_i^T x) + \beta_0 \tag{1.10}$$

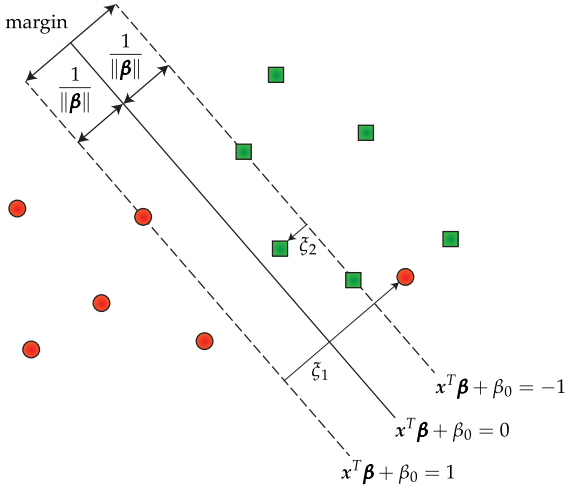


Figure 1.6: The geometric interpretation of a linear SVM.

Non-linear SVM with kernel

The SVM described so far allows finding linear boundaries in the input space. However, we can make the procedure more flexible by enlarging the original input space. The representation in feature space is obtained by the application of an appropriate function $\phi : x \in \mathcal{X} \longmapsto \phi(x) \in \mathcal{F}$. Consequently, we work with the samples: $(\phi(x_1), y_1), (\phi(x_2), y_2)..., (\phi(x_N), y_N) \in \mathcal{F} \times \mathcal{Y}$.

We can represent the optimization problem (primal form) as following:

$$\begin{aligned} &\min_{\beta} \ \frac{1}{2} \parallel \beta \parallel^2 + C \sum_{i=1}^{N} \xi_i \\ &s.t. \ y_i(\phi(x_i)^T \beta + \beta_0) \geq 1 - \xi_i; \ \ \xi_i \geq 0 \end{aligned} \tag{1.11}$$

and its dual form as:

$$\begin{aligned} &\max_{\alpha} \ \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} \alpha_i \alpha_{i'} y_i y_{i'} \langle \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_{i'}) \rangle_{\mathcal{H}} \\ &s.t. \ 0 \leq \alpha_i \leq C; \ \sum_{i=1}^{N} \alpha_i y_i = 0 \end{aligned} \tag{1.12}$$

The optimization solution α requires the computation of $\langle \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_{i'}) \rangle_{\mathcal{H}}$ for each pair of data in the set. One can use the kernel function $k(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle_{\mathcal{H}}$ to compute the inner product in feature space F directly from input space $\mathcal{X}$. One example of such "trick" is given in Eq. (1.5).

By solving the dual optimization problem, one obtains the nonlinear decision function

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}) \rangle + \beta_0 \ , \tag{1.13}$$

and in kernel function form, one obtains

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} y_i \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}) + \beta_0 \ . \tag{1.14}$$

Using kernel allows avoiding the computation of mapping $\phi(\boldsymbol{x})$ and the inner product in higher dimension space. In addition, the mapping function $\phi(\boldsymbol{x})$ is not always known *e.g.* Gaussian kernel, and the kernel trick allows one to compute implicitly the inner product without knowing the mapping function.

1.3.3 Large-scale learning for kernel methods

The kernel trick can compute implicitly the inner product in the feature space without mapping the data, which has been successful applied in a large range of problems from different domains. However it shows some limitations in the context of large-scale machine learning: methods operating on kernel matrix can hardly scale up *w.r.t.* training sample size, because of the calculation of kernel matrix (at least quadratic *w.r.t.* training sample size as shown in Eq. 1.12). This prevents them to be applied on large-scale learning problems [160].

Recently, techniques for kernel value approximation have been well investigated in the context of accelerating the training time in kernel methods [212], *e.g.*, the Nyström method and the Random Fourier features (RFF) technique. The Nyström method approximates the full kernel matrix by a low rank matrix computed with a subset of training examples, while the RFF technique [160, 161] is a data-independent method which is widely applied due to its efficient computation and approximation quality. The rational is to approximate the kernel by explicitly mapping the data (with basis functions as cosine and sine) into a low

dimensional Euclidean space, in which the inner product of the explicit features vector approximates the kernel value.

The advantages of RFF come with the fact that linear machine learning methods can be directly applied on the resulting vectors. While non-linear SVM using kernel yields a quadratic complexity *w.r.t.* training samples, linear SVM with efficient solver can reduce the complexity to linear. By adopting such a strategy, the empirical study in [125] shows the capability of training on large-scale image recognition problems.

1.3.4 Structured kernel

Nowadays, data are represented in a structured form in many real-world applications such as natural language processing [219], bioinformatics [128], chemoinformatics [81], XML trees in web mining [39, 41] and image processing [87]. Among the popular solutions to process such data are kernel methods, *e.g.* Support Vector Machine (SVM) [166]. Applying SVM on structured data requires either to vectorize the input data, or to define structured kernels. The latter option is to be preferred to benefit from the rich structure brought by strings, trees, or graphs.

For learning on structured data, various kernels have been proposed, which includes optimal assignment kernel [68, 105], alignment kernel [47], match kernel [19] and others [71]. However, the most standard way to construct a structured kernel is to follow the convolution kernel framework [88]. According to this framework, a kernel on a complex structure can be formed by tailoring simple kernels computed on its substructures.

Following the definition of the convolution kernel in [88, 169], let us define $x \in \mathcal{X}$ as a complex structure and x' as the parts of x by a relation R on the set $\mathcal{X}' \times \mathcal{X}$, where $R(x', x)$ is true if and only if x' are the parts of x. Given such a relation, we can define the decomposition $R^{-1} = \{x' : R(x', x)\}$. Suppose we have a kernel k on $\mathcal{X}'$ that measures the similarity $k(x', y')$ between the part x' and the part y' (note that x' and y' are often called substructures that can be extracted from complex structures x and y respectively). Then the convolution kernel is defined as:

$$K(x, y) = \sum_{\{x'\} \in R^{-1}(x)} \sum_{\{y'\} \in R^{-1}(y)} k(x', y') \ , \tag{1.15}$$

where if the kernel $k : \mathcal{X}' \times \mathcal{X}' \to \mathbb{R}$ is a positive semidefinite kernel, $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is also positive semidefinite.

Following such a definition, the key idea is to define the appropriate substructures that can decompose the complex structure. However, there is no universal substructure for structured data, and the available options are numerous (see [168] for the case of trees). The selection of the appropriate substructure thus depends on the kind of data and application

that are considered. To drive this selection, one can keep in mind that according to [98], a "good" kernel depends on both its effectiveness and expressiveness power. The former refers to how good a substructure can represent the whole structure, while the later refers to the computational complexity required when calculating such a kernel. Most often the appropriate kernel is the result of a trade-off between expressiveness and effectiveness. If the substructure is too simple, the kernel might not be able to capture the characteristics of the underlying data. Conversely, complicated substructures might lead to intractable kernel computation schemes.

Nevertheless, the capabilities of taking directly into account structured data have motivated various attempts to define new kernels, *e.g.* in bioinformatics [131], physics [63], language processing [175], or even image analysis. Indeed, many solutions have been introduced recently to perform machine learning on graph-based image representations [8, 87, 200, 65, 177, 179, 57]. Such representations are usually based on region-adjacency graphs that are built from a prior segmentation of the input image, where the vertices of the graph encode the regions while the edges encode the relations between neighboring regions [8, 87, 200].

Inspired by the literature in kernel design for various types of structures, especially for graph-based image representation, we explore in this thesis how to design a structured kernel that make possible learning from hierarchical image representations.

1.4 Conclusion and organization of the manuscript

1.4.1 Motivations and contributions

In this chapter, we first briefly reviewed the trends in remote sensing image classification in Sec. 1.1, where one may notice that including the spatial information is one of the key concepts for improving classification accuracy. It can help increasing smoothness for pixel-wise classification, and providing crucial patterns for tile classification. One way to reveal spatial information relies on hierarchical image representations, where objects-of-interest can be revealed at various scales and topological relationships among them are modeled through the hierarchy. Indeed, such representations have been adopted with various applications in the remote sensing community, as reviewed in Sec. 1.2.

This thesis aims to link machine learning techniques and hierarchical image representations. Our goal is to take into account spatial information presented in an image through a hierarchical organization of objects. Such a hierarchy of objects can be represented with a tree structure, where the nodes represent the objects-of-interest and edges model the hierarchical relationships among them. Our objective is to exploit machine learning techniques to take into account these specific representations of data and discover the meaningful patterns

that can lead to good classification results.

In the literature, one of the most standard ways to learn on structured data is to design a structured kernel. It can directly take the structures as input and benefit from various successful kernel-based machine learning methods. Such a scheme is especially well-adopted in the domain of bioinformatics, natural language processing and chemistry, where the data to be handled are often coming in a structured form. In addition, the kernel methods, especially SVM, have been well established in remote sensing [27]. These various aspects motive us for exploring kernel-based learning on hierarchical image representations.

In addition, we often face a large amount of data in the remote sensing community. Meanwhile, advanced techniques such as RFF enable us to apply kernel methods in a large-scale context. This motivate us to develop scalable methods to enable the application of proposed kernels to remote sensing image classification.

We introduce in this thesis a structured kernel called (S)BoSK (Bag of Subpath Kernel and its Scalable version) for capturing the hierarchical relationships between nodes of a tree. It can be viewed as an instance of the convolution kernel relying on the extraction of subpath substructures. The main applications focus on remote sensing image classification: (S)BoSK operates on paths, thus allowing modeling the context of a pixel (leaf of the hierarchical representation) through its ancestor regions at multiple scales; (S)BoSK also works on trees, that makes the kernel able to capture the composition of an object (top of the hierarchical representation) and the topological relationships among its subparts; relying on (S)BoSK, we also introduce a novel multi-source classification approach that performs classification directly from a hierarchical image representation built from two images of the same scene taken at different resolutions, possibly with different modalities.

1.4.2 Organization

The rest of the thesis is organized as follows:

- **Chapter 2** starts introducing the main contributions of this thesis from a methodological point of view. We present a structured kernel called Bag of Subpaths kernel (BoSK) for data with an unordered tree or path structure, and equipped with numerical features. Both exact computation based on iterative scheme and approximated computation based on Random Fourier Features are provided in different contexts: while BoSK can be efficiently computed for small structures and small training data size, its scalable version SBoSK is more suitable to large-scale learning context with a large number of available training samples or/and large structures.
- **Chapter 3** presents the first application of (S)BoSK on the path structure for pixel-wise image classification. The path structure represents the spatial context of a pixel in the

hierarchical representations, where the nodes in each path start from the pixel and continue with its ancestral regions at multiple scales from fine to coarse. Relying on such a structure, (S)BoSK can take into account the contextual information for each pixel by exploiting the regions from different scales and the hierarchical relationships among them. Evaluations on different datasets indicate the superiority of (S)BoSK over other spatial spectral pixel-wise classification techniques.

- **Chapter 4** introduces the second application of (S)BoSK on the tree structure for sub-image/tile classification. The root of a tree structure is the tile and rest of the nodes are the subregions at multiple scales organized hierarchically. Assessing inputs as tree structures, (S)BoSK allows considering spatial decomposition of a tile through its sub-regions and relationships among them. Evaluations on different datasets show that (S)BoSK can surpass other state-of-the-art decomposition-based techniques.

- **Chapter 5** presents a novel multi-source and multi-resolution image classification method that combines (through a hierarchical representation) two images taken from different resolutions and sensors over the same area. The coarser levels of the hierarchy built from the lower resolution image can reveal contextual information, while the finer levels are constructed from the higher resolution image and are used to model the spatial decomposition. Two (S)BoSK are then employed to perform machine learning directly on the constructed hierarchical representation, aiming at combining both contextual and decomposition information into a unique classification scheme.

- **Chapter 6** provides some conclusions from the work presented herein, along with considered improvements closely related to the proposed methods, as well as some perspectives on future research directions.

1.4.3 List of publications

The following publications are based on the research presented in this thesis:

1. Yanwei Cui, Laetitia Chapel, and Sébastien Lefèvre. "A subpath kernel for learning hierarchical image representations". In: *International Workshop on Graph-Based Representations in Pattern Recognition*. 2015, pp. 34–43. DOI: 10.1007/978-3-319-18224-7_4

2. Yanwei Cui, Laetitia Chapel, and Sébastien Lefèvre. "Combining multiscale features for classification of hyperspectral images: a sequence based kernel approach". In: *International Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*. 2016. URL: http://arxiv.org/abs/1606.04985

3. Yanwei Cui, Sébastien Lefèvre, Laetitia Chapel, and Anne Puissant. "Combining multiple resolutions into hierarchical representations for kernel-based image classification". In: *International Conference on Geographic Object-Based Image Analysis*. University of Twente, Enschede, The Netherlands, 2016. DOI: 10.3990/2.372

4. Yanwei Cui, Laetitia Chapel, and Sébastien Lefèvre. "Scalable bag of subpaths kernel for learning on hierarchical image representations and multi-source remote sensing data classification". In: *Remote Sensing, Special Issue Advances in Object-Based Image Analysis—Linking with Computer Vision and Machine Learning* 9.3 (2017). DOI: 10.3390/rs9030196

Chapter 2

Scalable Bag of Subpaths Kernel (SBoSK) for numerical features

Contents

In the previous chapter, we have presented why learning on hierarchical image representations was an open challenge that we address in this thesis.

This chapter describes our main contributions from a methodological point of view. We present a structured kernel dedicated to unordered trees with numerical features. This chapter concentrates on presenting the mathematical details of the proposed kernel. Its applications to machine learning on hierarchical image representations are further detailed in the following chapters.

2.1 Introduction

Data are often represented in a structured form in many real-world applications. The structure helps encoding the internal relations among elements. For instance, an XML tree encodes the hierarchical relationships among different elements presented in a document. A molecule can be analyzed as a graph structure that represents the structural formula of atoms. In image processing, it is popular to describe the content of an image through structured data, where image regions are presented in a set, and are organized through structures linking different regions together. Successful examples of such representations are region adjacency graph [87], or tree structures [17].

In the context of machine learning, conventional techniques often require the data to be in a vector form. In order to learn on structured data, one popular strategy consists in designing meaningful structured kernels (*i.e.* kernels built on structured data), and feed them into kernel-based machine learning methods such as SVM. Such kernels have been successfully applied in various domains as they are capable to extract discriminative features directly from the structures. Among popular approaches for designing kernels, the most standard way to construct a structured kernel is to follow the convolution kernel framework [88]. According to this framework, computing a kernel on a complex structure can be achieved by summing up the kernels built on its substructures. In the literature, various substructures have been used (see [168] for the case of trees) and the selection of the appropriate substructure depends on the kind of data and application that are considered.

In our context *i.e.* designing a kernel that can learn on hierarchical image representations, several critical aspects have to be taken into account.The importance of these aspects will be illustrated in the following chapters with concrete examples.

- **Unordered tree**: we concentrate on kernels that can handle unordered trees (through which hierarchical image representations are often modeled). Such kernels should be able to capture the hierarchical relationships (*i.e.* parent-children relation) among the nodes.

- **Numerical features**: another important property in hierarchical image representations is that each node represents a region, and that attributes of a region, *e.g.* color or size, are in general numerical features. This actually differs from many other domains where nodes are labeled by a fixed number of symbols.

- **Robustness to structure distortion and noise**: hierarchical representations heavily rely on the adopted construction techniques. These techniques build the tree in an unsupervised way, which tree structure might vary due to complexity of image contents, presence of the noise, or undesired regions grouped together. Thus the resulting struc-

tures are less strict than the one in other domains such as chemoinformatics. This should be taken into account when designing the kernel.

- **Complexity**: the adopted kernel should be efficient and scalable. Unlike other domains, such as chemoinformatics, or nature language processing, where the data structures are relatively small, hierarchical representations often have a large number of nodes, and attribute of each node might be up to thousands of dimension. In addition, some image classification problems in literature might possess a large number of available training samples, complexity issues are then critical.

To address all the aspects mentioned above, we propose a structured kernel based on the concept of subpath. It works on vertical hierarchical relationships among nodes in the structured data, with nodes equipped with numerical features.

For its computation, we propose an iterative approach with a quadratic complexity *w.r.t.* the size (*i.e.* number of nodes) of structured data, and a quadratic complexity *w.r.t.* number of training samples. It is efficient when dealing with small structure size and limited number of training samples, and we call it BoSK (for Bag of Subpaths Kernel).

In addition, when running on large-scale datasets, we propose to compute the kernel approximately by explicit mapping the kernel into randomized low dimensional feature spaces using Random Fourier Features. This approximation yields a linear complexity *w.r.t.* size of structured data, and a linear complexity *w.r.t.* number of training samples. Therefore, the resulting approximation scheme makes the kernel applicable for large-scale real world problems. We call it Scalable Bag of Subpaths Kernel (SBoSK).

The chapter is organized as follows: we give a brief review in Sec. 2.2 on learning on structured data, with a particular focus on convolution kernels and large-scale structured kernels. Then we introduce BoSK in Sec. 2.3, with its scalable version SBoSK in Sec. 2.4. In the end, we summarize the chapter in Sec. 2.5.

2.2 Related work

2.2.1 Learning on structured data

Various approaches have been proposed in order to perform machine learning on structured data. In this section, we give a brief review of different directions that can be found in the literature.

Defining distance measures has already existed for a long time for learning on structured data [182, 163, 69, 187]. Among various distances, the edit distance is considered as one of the most well established frameworks in pattern recognition and classification [14, 69]. It

is defined as the minimum cost of operations needed to transform one structure into another. Such operations include insertion, deletion, or substitution, where each of them is associated with a cost. Making two structures isomorphic with a minimum cost provides an intuitive way of defining the similarity between two structures. Various methods have been successfully used for different structure types, *e.g.* sequence [163], tree [14] and graph [69]. However, it is still difficult to be applied in our context for the following reasons.

The first challenge is to define a reasonable cost for each type of edit operations. As the edit distance is computed based on these costs, only appropriate cost definition can lead to a good performance in recognition and classification tasks. However, the definition of cost depends highly on the particular problems at hand and it is in general fixed empirical [69].

In addition, the computed distances are commonly used in the distance-based classification framework, such as K-nearest neighbors algorithm. Other powerful classification frameworks such as kernel methods can not be applied [57]. An attempt has been made to apply SVM with kernel function based on edit distance in [144], it achieved higher classification accuracy compared to the traditional nearest-neighbor classifier. However, the validity of the kernel using edit distance cannot be established.

Finally, the computation of edit distance requires finding the minimum cost, and it is in general computationally expensive for complex structures such as for trees or graphs [14, 69, 6]. In case of unordered tree, such a computation becomes NP-hard [14, 6].

In order to learn on structured data, another direction is to define kernels and apply kernel-based machine learning algorithms. In this direction, the most standard way to construct valid kernels is to follow the convolution kernel framework [88]. It states that computing a kernel on a complex structure could be achieved by summing up kernels on its substructures. Following this framework, a large number of structured kernels have been proposed under different decompositions [201]. As most works in the literature focus on designing structured kernels under the well-established convolution kernel framework, we will briefly review them in the following section.

Before going to the details of convolution kernels, let us note that there exists another class of kernels based on decomposing structured data into substructures, called optimal assignment kernel [68, 105]. Instead of adding up all pairwise similarities between all their parts, it is computed on optimal bijection between the substructures, meaning that only the optimal matches (the highest similarity values) will contribute to the overall kernel value. Thanks to this aspect of assigning the parts of one objects to the parts of the other, experiments with some symbolic structured datasets in [105] show an improvement of classification accuracy compared to their convolution-based counterparts relying on the same substructures. However, its relevance for structured data with numerical features, especially for images, remains to be demonstrated. Besides, the derived similarities are not necessarily positive definite [105, 199], thus not a valid kernel. One recent study has shown its valid

condition can be achieved with pairing with a particular class of atomic kernels [105], while in the case of Gaussian kernel, the optimal assignment kernel is, unfortunately, not positive definite [199]. Due to the aforementioned reasons, we follow the mainstream of designing kernel in the convolution kernel framework.

2.2.2 Convolution kernels

Various types of convolution kernel have been proposed to cope with tree structures [168], differing in the selection of substructures for specific problems. The subtree kernel [202] counts the common subtree between two trees, while the subset tree kernel [39] relies on the richer substructure of the subset of subtree. Some extensions for the subset tree kernel have been proposed with partial matching [137] and elastic matching [99] strategies. Kernels coping with relative nodes positions have been also proposed in [2, 1]. The reader will find additional kernels and appropriate references in a survey from [168]. While these existing works are numerous, it is important to notice that most of them deal with ordered trees.

In the case of unordered trees, the most popular solution is to rely on the concept of subpath [102, 101] that has been identified as an appropriate substructure to ensure satisfying levels of expressiveness and effectiveness. Furthermore, the survey on tree kernels from [168] was considering it as the only suitable substructure for unordered trees. Although more expressive substructures such as subtrees have been proposed for specific constrained unordered tree [85, 92], their adaptation to arbitrary unordered trees still faces computational issues [98].

In fact, path substructures are popular when dealing with complex structures such as graphs [3, 100, 20], mainly because of the good balance between the effectiveness and the computational complexity [72]. Many successful graph kernels have been proposed in the literature using paths, including the marginalized graph kernel [100] using all possible paths, the shortest path graph kernel [20], or the graph kernel using paths with maximum considered lengths [21].

Moreover, let us recall that we focus here on structured data equipped with numerical features. It requires the definition of a kernel on substructures capable of taking into account numerical values. The above mentioned kernels and their efficient computational algorithms rely on structured data containing symbolic attributes, while structured data where elements are described with numerical features have motivated various attempts to define new kernels or adapting the existing ones to their numerical version in various domains *e.g.* in bioinformatics [131], physics [63], language processing [175], or even image analysis [87].

In the domain of image analysis, region-adjacency graphs can be built from a prior segmentation of the input image to reveal detailed content of image, where the nodes of the graph encode the regions and the edges encode the relations between neighboring regions.

With such a representation, the marginalized graph kernel with numerical features version has been proposed in [8], and several walk-based graph kernel were used in [87, 200, 65, 110]. Another graph structure commonly adopted when dealing with image data is the skeleton graph used for shape recognition. Similarly, dedicated graph kernels such as the marginalized graph kernel [177, 179] or other graph kernel based on paths [57] have been proposed.

2.2.3 Large-scale structured kernel

The major issue of structured kernels is their computational complexity. This limits their application to small data volume, and small structure size. Previous works successfully bring down the kernel computational complexity to be linear such as [202, 101], with symbolic data type. However, in case of data equipped with numerical features, it is often reported as quadratic complexity such as [47] for sequence data, and even worse for graph kernel which yields polynomial time with higher order [87, 128, 20]. Although some of the structured kernels are entitled scalable kernel in the literature, such as [63], their complexity is still polynomial, preventing them to be used in the context of large structures. As each pair of nodes between two structures has to be at least compared once to compute the overall kernel value, structured kernels on numerical data always yield at least a quadratic complexity. Such high complexity techniques for structured kernels are still in use nowadays [70].

A recent survey of graph kernel [104, 106] indicates that recent graph kernels do not employ the kernel trick anymore but rather compute an explicit feature map. Although implicit kernel computation is considerably faster than explicit computation when dealing with rather small structure size, recent successes on scalable structured kernel suggests that explicit feature vector is the key concept of scalability [167]. Indeed, such a strategy enables the efficient computation of structured kernel, allowing handling thousands of nodes. Successful examples *e.g.* graphlet kernel [167], treelet kernel [73] are based on explicitly counting the common predefined unlabeled substructures, and [104] derives efficient explicit mapping for several well known graph kernels with symbolic node features.

In case of numerical features, the strategy of using explicit feature maps for structured kernels begins to be adopted. One possible direction is to pass the numerical features into symbolic ones, so that the previous proposed scalable kernels on symbolic node features can be applied. Following this direction, binning [145] and hashing function [136] have been proposed for structured kernels. Very recently, approximated kernel computation based on explicit feature maps has been proposed in [106].

Meanwhile, in the context of large-scale machine learning, techniques for kernel value approximation have been well investigated in order to reduce the training time in kernel methods [212] The two main techniques are Nyström method and Random Fourier Fea-

tures. Nyström method approximates the full kernel matrix by a low rank matrix computed with a subset of training examples. Although it has been successfully applied in large-scale machine learning context, it still requires the kernel matrix computation and this might be time consuming if a large number of subsamples is needed or pairwise kernel value computation is slow, such as in the case of structured kernels. RFF technique [160, 161], however, is a data independent method, which is widely applied due to its efficient computation and approximation quality. The idea is to approximate the kernel by explicitly mapping the data (with basis functions as cosine and sine) into a low dimensional Euclidean space, in which the inner product of explicit features vector approximate the kernel value. By adopting such a strategy, empirical study in [125] shows the capability of training on large-scale image recognition problems. In addition, RFF have been applied in order to reduce the computational complexity for match kernel that is computed between two sets of local descriptors (*e.g.* SIFT) extracted from images [19].

To sum up, the literature related to learning on structured kernels is very rich, among which convolution kernel framework is the most standard way to design kernels for structured data. Under this framework, path substructures have been successfully considered for complex structures such as trees and graphs due to their satisfying levels of expressiveness and effectiveness. The subpath kernel, which relies on paths and nodes, can be considered as one successful example for unordered tree. While it has been originally proposed for symbolic node features, its numerical version can be inspired by numerous successful adaptations in the case of graph kernels, especially for image analysis. In addition, the recent attempts on structured kernel computation using explicit feature maps motivate us to adopt kernel approximation strategy to ensure the scalability of the kernel.

2.3 Bag of Subpaths Kernel

2.3.1 Basic definitions and notations

Before going through the details of the bag of subpaths kernel, let us first establish some basic definitions and notations.

A graph $\mathcal{G}$ is defined by the tuple $\mathcal{G} = (N, E)$, where nodes N are a finite set of elements, and edges E are the pairwise relationships between those nodes. The structure size $|\mathcal{G}|$ is the number of nodes in $\mathcal{G}$. A node $n_i \in N$ is described by its features, which can be either symbolic, or numerical. In case of numerical, they can be defined as a d-dimensional vector $\boldsymbol{x}_{n_i} \in \mathbb{R}^d$. An edge e_{n_i,n_j} is the connection between n_i and n_j. If such connection in graph has a direction $e_{n_i,n_j} \neq e_{n_j,n_i}$, we call it directed graph, otherwise, it is defined as an undirected graph.

A path $\mathcal{P}$ in $\mathcal{G}$ is a sequence of nodes $\mathcal{P} = (n_{(1)}, n_{(2)}, \ldots, n_{(p)})$, where $n_{(i)} \in \mathcal{G}$ and each

consecutive pair $\{n_{(i)}, n_{(i+1)}\}$ is linked by an edge in $\mathcal{G}$, (i) denotes the relative position of nodes in a path and p being the length of path. A cycle is a path where $n_{(1)} = n_{(p)}$ and other nodes $\{n_{(2)}, \ldots, n_{(p-1)}\}$ are distinct.

A graph $\mathcal{G}$ is connected if every pair of nodes has a path between them, and is acyclic when $\mathcal{G}$ does not contain any cycle. If the graph is connected and acyclic, it is defined as a tree $\mathcal{T}$.

We refer a tree $\mathcal{T}$ as a directed rooted tree with one designated node called root and all edges are directed away from root. Regarding the pairwise relationship in a tree $\mathcal{T}$, if there is an edge e_{n_i,n_j} connecting two nodes n_i and n_j, we call n_i the parent of n_j, and n_j a child of n_i. The leaves of tree are the nodes without any child. An ordered tree is one in which the children of each node follow a specified ordering, otherwise it is called an unordered tree.

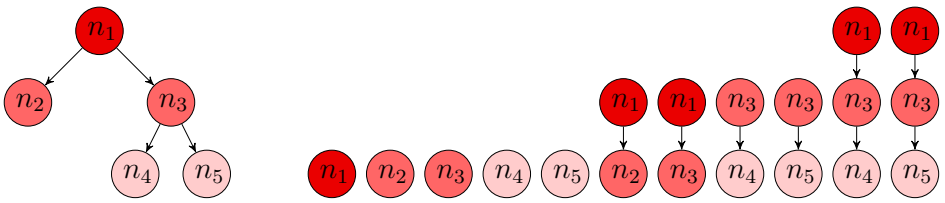
In this thesis, $\mathcal{G}$ is used for representing a hierarchical representation, where nodes stand for the image regions (objects-of-interest) in the hierarchy, and edges encode the pairwise relationships among regions. Depending of the orientation of the edges, we will refer to $\mathcal{G}$ as a directed rooted unordered tree $\mathcal{T}$ (or tree for short) when it is read away from the root (top-down); when read from the leaves towards the root (bottom-up), it can be decomposed as a set of paths $\mathcal{P}$. In such region hierarchy, given two regions: n_i at higher level and n_j at lower level *i.e.* $n_j \subseteq n_j$, we call n_i an ancestor of n_j (or an ancestral region of n_j), and n_j a descendant of n_i (or a subregion of n_i). See Fig. 2.1 for an example of $\mathcal{T}$ and $\mathcal{P}$.

2.3.2 Kernel definition

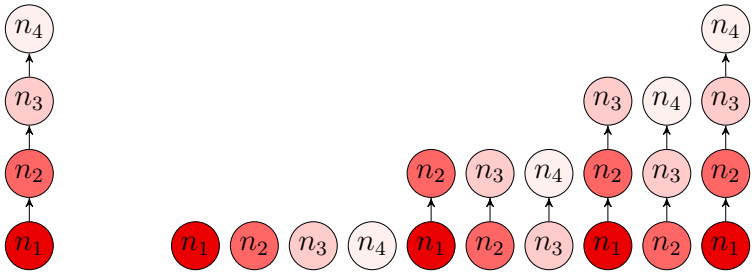
In order to capture the vertical hierarchical relationships between the nodes, we decompose either $\mathcal{T}$ or $\mathcal{P}$ as a set of substructures called subpaths. A subpath is defined as the path connecting a node to one of its descendants (*resp.* ancestors) in $\mathcal{T}$ (*resp.* $\mathcal{P}$); the set of subpaths also includes individual nodes. Let us denote a subpath by $s_p = (n_{(1)}, n_{(2)} ..., n_{(p)})$, $s_p \in \mathcal{G}$ with length p. Examples of a tree and a path, together with their sets of subpaths, are shown in Fig. 2.1.

Subpath substructure has been proposed in [102, 101] for unordered tree with nodes equipped with symbolic features. It has been proved to be one of the most effective substructures for building tree kernels, and is considered as one suitable substructures in terms of complexity when applied on unordered trees [168]. However, the concept and the kernel computation algorithms proposed in [102, 101] are based on counting common substructures between two trees, which are only applicable for symbolic data.

For numerical features, the concept of counting common subpaths is required to be changed since strict identity between subpaths (and their respective node features) does not generally occur. BoSK replaces it by using a kernel that measures the similarity between



(a) A tree $\mathcal{T}$ and all its subpaths s_p.



(b) A path $\mathcal{P}$ and all its subpaths s_p.

Figure 2.1: Examples of structured data that can be extracted from hierarchical image representations, a tree $\mathcal{T}$, a path $\mathcal{P}$ and their subpaths.

two bags of subpaths embedded in two structures $\mathcal{G}, \mathcal{G}'$. More specifically, the definition of BoSK between $\mathcal{G}$ and $\mathcal{G}'$ is written as:

$$K(\mathcal{G}, \mathcal{G}') = \sum_{p=1}^{P} \mu_p \sum_{s_p \in \mathcal{G}} \sum_{s'_p \in \mathcal{G}'} K(s_p, s'_p) \,, \tag{2.1}$$

where the first sum is defined over the different lengths of subpaths, with P being the maximum subpath length extracted from $\mathcal{G}$. The second and third sums allow the computation of the kernel over all pairs of subpaths in $\mathcal{G}$ and $\mathcal{G}'$ (note that only the matching of subpaths of the same length is permitted), which is further weighted by μ_p (different options of weighting are proposed and analyzed in the next section). The kernel $K(s_p, s'_p)$ between two subpaths s_p and s'_p is defined as the product of atomic kernels computed on pairs of nodes $k(n_{(t)}, n'_{(t)})$ of the subpaths:

$$K(s_p, s'_p) = \prod_{t=1}^{p} k(n_{(t)}, n'_{(t)}) \,. \tag{2.2}$$

One might notice that if $k(n_{(t)}, n'_{(t)}) = \delta_{n_{(t)}, n'_{(t)}}$, the Kronecker delta function that measures the identicalness of $n_{(t)}$, $n'_{(t)}$, then $K(s_p, s'_p) = 1$ iff the two subpaths s_p, s'_p are identical, otherwise $K(s_p, s'_p) = 0$. BoSK in Eq. (4.1) will then count the common subpaths between two structures. Therefore, we can state that BoSK is a generalization of the subpath kernel proposed in [102, 101].

2.3.3 Kernel weighting and normalization

The definition of the subpath kernel given in Eq. (4.1) involves a parameter μ_p that weights the different subpath lengths. Such weights can change the contribution of kernels computed on each individual subpath length, allowing one to incorporate prior knowledge into kernel construction *e.g.* limit the contributions of longer subpaths by setting a smaller weight on larger length p. Several weighting schemes can be found in the literature [101, 39, 87, 202]:

- **Constant weights:** $\mu_p = 1$ for all p, leading to a constant weighting for all lengths of subpaths. It is a common strategy of weighting equally among different lengths.
- **Exponential weights:** $\mu_p = \lambda^p$ with $\lambda \in (0, 1)$, an exponentially decaying weight *w.r.t.* the length of the subpaths. This will downweight the contributions of larger subpaths. The strategy is commonly adopted in recursive computation, as only λ needs to be multiplied by the atomic kernel in Eq. (2.2) [39, 87].
- **Maximum considered length:** $\mu_p = 1$ for all $1 \leq p \leq q$, considering only a limited number of subpath patterns. This also limits the contributions of larger subpaths, and has been commonly adopted when dealing with an enumeration of all possible substructures, as less substructures need to be considered [101, 202].

A well known issue of structured kernels is that their value highly depends on the size of the structure. This comes from the fact that the overall kernel value relies on summing up all the kernel values on substructures (see in Eq. (4.1)): the more substructures one can extract, the larger the kernel value is. In the literature, this problem can be mitigated using kernel normalization strategy [39, 166]:

$$\begin{aligned} K^*(\mathcal{G}, \mathcal{G}') &= \left\langle \frac{\phi(\mathcal{G})}{\|\phi(\mathcal{G})\|}, \frac{\phi(\mathcal{G}')}{\|\phi(\mathcal{G}')\|} \right\rangle = \frac{\langle \phi(\mathcal{G}), \phi(\mathcal{G}') \rangle}{\sqrt{\langle \phi(\mathcal{G}), \phi(\mathcal{G}) \rangle} \sqrt{\langle \phi(\mathcal{G}'), \phi(\mathcal{G}') \rangle}} \\ &= \frac{K(\mathcal{G}, \mathcal{G}')}{\sqrt{K(\mathcal{G}, \mathcal{G})} \sqrt{K(\mathcal{G}', \mathcal{G}')}}, \end{aligned} \tag{2.3}$$

where $\phi(\mathcal{G})$ is the mapping introduced in Sec. 1.3.1. By adopting such kernel normalization, we ensure the overall kernel value being always in $(0, 1]$, with $K^*(\mathcal{G}, \mathcal{G}) = 1$.

2.3.4 Efficient computation for BoSK

We propose here an unified algorithm for computing BoSK on tree and path structures. The basic idea is to iteratively compute the kernel on subpaths $K(s_p, s'_p)$ of length p using previously computed kernels on the subpaths $K(s_{p-1}, s'_{p-1})$ of length $p - 1$. The atomic kernel $k(n_i, n'_j)$ between each pair of nodes $(n_i \in \mathcal{G}, n'_j \in \mathcal{G}')$ thus needs to be computed only once.

We define a three-dimensional matrix M of size $|\mathcal{G}| \times |\mathcal{G}'| \times P$, where each element $M_{i,j,p}$ is computed iteratively as:

$$M_{i,j,p} = k(n_i, n'_j, p)(M_{\text{parent}(n_i),\text{parent}(n'_j),p-1}) \ , \tag{2.4}$$

where $M_{0,0,0} = M_{i,0,0} = M_{0,j,0} = 1$ by default, $\text{parent}(n_i)$ refers as parent index of the node n_i. The parent index of each node can be constructed by presenting the tree as a sequence of nodes with a pre-order depth-first traversal algorithm [93]. By convention, the parent index of the root of a tree is 0. In case of the path structure $\mathcal{P}$, the parent index $\text{parent}(n_i)$ is simply the index of the node n_{i-1}.

The overall kernel value is then computed as the sum of all the matrix elements:

$$K(\mathcal{G}, \mathcal{G}') = \sum_{i=1}^{|\mathcal{G}|} \sum_{j=1}^{|\mathcal{G}'|} \sum_{p=1}^{P} M_{i,j,p} \ . \tag{2.5}$$

The overall complexity of BoSK is bounded by the computation of the three-dimensional matrix M, which yields $O(|\mathcal{G}|\ |\mathcal{G}'|\ d)$.

2.4 Scalable Bag of Subpaths Kernel

2.4.1 Implicit v.s. explicit computation

Computing kernel using explicit feature maps allows the use of kernel methods in the large-scale machine learning context [155, 140], thanks to some recent developments that make it possible to train a linear method with a linear complexity *w.r.t.* training sample size n, instead of a quadratic one in kernel methods [59]. As nonlinear kernel computation can be seen as an inner product operation in the feature space, explicit kernel computation has been extensively studied [160, 161, 198].

Such an explicit kernel computation strategy has also been used in the convolution kernel framework. Many of the recent proposed kernels compute the feature maps explicitly to ensure the scalability of kernel [167, 73, 104, 106].

[106, 104] have summarized and illustrated the advantages of using explicit computation in the context of structured kernel. Given a kernel matrix of $n \times n$ to be computed using structured kernel, let us define T_K the pairwise implicit kernel computation time, T_ϕ the computation time for explicit feature map and T_{dot} the computation time for dot product between two vectors. Then we have $O(n^2 T_k)$ for implicit kernel matrix computation, while explicit kernel computation on feature space needs $O(nT_\phi + n^2 T_{dot})$ to compute the same kernel matrix — it maps each structured data into a low dimensional Euclidean space with $O(nT_\phi)$, then computes the dot product between two embedded vector with $O(n^2 T_{dot})$. It should be

noted that, in general, when dealing with large-scale training samples n, one rather chooses linear learning algorithms such as [59] instead of kernel methods. In this case, the kernel matrix no longer needs to be computed, so that $n^2 T_{dot}$ can be avoided.

In the previous section, we introduced an efficient algorithm that can compute the T_k in $O(|\mathcal{G}|\,|\mathcal{G}'|\,d)$. Therefore, the overall computation needs quadratic complexity *w.r.t.* structure size, and *w.r.t.* number of training samples n. Such a computation scheme is efficient for small-scale datasets.

In order to address complexity issue in the large-scale context, computation based on explicit feature maps appears attracting. By adopting an explicit computation, the overall complexity mainly depends on the embedding of the structured data *i.e.* $O(nT_\phi)$.

We present here the explicit computation for BoSK. The conventional choice of the atomic kernel $k(\cdot)$ is the Gaussian kernel. In that case, the kernel $K(s_p, s'_p)$ can be written as:

$$\begin{aligned} K(s_p, s'_p) &= \prod_{t=1}^{p} \exp(-\gamma ||\boldsymbol{x}_{n_{(t)}} - \boldsymbol{x}_{n'_{(t)}}||^2) \\ &= \exp(-\gamma ||\boldsymbol{x}_{s_p} - \boldsymbol{x}_{s'_p}||^2) \\ &= \langle \phi(\boldsymbol{x}_{s_p}) \,,\, \phi(\boldsymbol{x}_{s'_p}) \rangle_{\mathcal{H}} \,, \end{aligned} \tag{2.6}$$

where $\boldsymbol{x}_{s_p} = [\boldsymbol{x}^T_{n_{(1)}}, \boldsymbol{x}^T_{n_{(2)}}, \cdots, \boldsymbol{x}^T_{n_{(p)}}]^T \in \mathbb{R}^{pd}$ is the numerical feature of subpath s_p, being the concatenation of the features of the nodes. Following the definition of a kernel function, one can write $K(s_p, s'_p)$ in the inner product form in a Hilbert space $\mathcal{H}$ as $\langle \phi(\boldsymbol{x}_{s_p}), \phi(\boldsymbol{x}_{s'_p}) \rangle_{\mathcal{H}}$, where $\phi(\cdot)$ is the mapping function for the Gaussian kernel [166].

By using the explicit mapping function for the Gaussian kernel, BoSK can be rewritten as follows:

$$\begin{aligned} K(\mathcal{G}, \mathcal{G}') &= \sum_{p=1}^{P} \sum_{s_p \in \mathcal{G}} \sum_{s'_p \in \mathcal{G}'} \langle \phi(\boldsymbol{x}_{s_p}), \phi(\boldsymbol{x}_{s'_p}) \rangle_{\mathcal{H}} \\ &= \sum_{p=1}^{P} \langle \sum_{s_p \in \mathcal{G}} \phi(\boldsymbol{x}_{s_p}), \sum_{s'_p \in \mathcal{G}'} \phi(\boldsymbol{x}_{s'_p}) \rangle_{\mathcal{H}} \,. \end{aligned} \tag{2.7}$$

The explicit mapping function $\phi(\cdot)$ hence brings down the quadratic computational complexity of $K(\mathcal{G}, \mathcal{G}')$ to a simple inner product computation with a linear complexity, as the double sum operation changes to a simple sum computed independently for each subpath.

In the following section, we adopt Random Features Features for approximation of mapping function $\phi(\cdot)$ of atomic Gaussian kernel, then derive the scalable version of BoSK (called SBoSK) that computes the structured kernel in a linear time *w.r.t.* structure size and *w.r.t.* number of training samples n.

2.4.2 Ensuring scalability using Random Fourier Features

The definition of the explicit mapping function $\phi(\cdot)$ is crucial for bringing down the complexity for structured kernel but it is unknown. Approximations have been well investigated in the context of accelerating the training of kernel machines [212]. Here, we consider Random Fourier Features (RFF) [160, 161]: the idea is to approximate the kernel by explicitly mapping the data into a low dimensional Euclidean space, where the inner product of the mapping function $z(\cdot)$ approximates the kernel value:

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x'}) \rangle_{\mathcal{H}} \approx z(\boldsymbol{x})^T z(\boldsymbol{x'}) \ . \tag{2.8}$$

The approximation function $z(\cdot)$ [180] for the Gaussian kernel can be written as:

$$z(\boldsymbol{x}) = \sqrt{\frac{2}{D}} \begin{bmatrix} \cos(\omega_1 \boldsymbol{x}) \\ \sin(\omega_1 \boldsymbol{x}) \\ \cdots \\ \cos(\omega_{\frac{D}{2}} \boldsymbol{x}) \\ \sin(\omega_{\frac{D}{2}} \boldsymbol{x}) \end{bmatrix}, \quad \omega_i \overset{iid}{\sim} \mathcal{N}(0, 2\gamma) \ , \tag{2.9}$$

where D is the dimension of the RFF vector, and the weight vector ω_i is drawn from a Gaussian distribution of mean 0 and variance 2γ, γ being the bandwidth parameter of the Gaussian kernel. By using $z(\cdot)$ to approximation the Gaussian kernel, one can obtain an exponentially fast convergence with D [160]. This means that the higher the RFF dimension D, the better the kernel approximation quality. However, a larger D increases the computational complexity. Therefore, in general, D is fixed empirically (depending on the problem at hand) as a trade-off between the quality of kernel approximation and computational complexity [160, 125].

However, the trade-off between the quality of kernel approximation and computational complexity.

We can then write:

$$K(\mathcal{G}, \mathcal{G}') = \tau(\boldsymbol{s})^T \tau(\boldsymbol{s'}) \ , \tag{2.10}$$

where the set of vectors encoded into the feature space for each subpath s_p are aggregated inside a single vector $\tau(\boldsymbol{s}) = [\sum_{s_1 \in \mathcal{G}} z(\boldsymbol{x}_{s_1})^T, \cdots, \sum_{s_p \in \mathcal{G}} z(\boldsymbol{x}_{s_p})^T]^T$.

2.4.3 Kernel normalization

We propose to adopt a L_2 normalization strategy dedicated to structured kernel using RFF as it is commonly used as a preprocessing step in computer vision community before applying

linear kernel [154, 186]. To do so, we perform L_2 normalization on the RFF vector for each subpath length before concatenating the normalized vectors together:

$$\tau(\boldsymbol{s}) = \frac{1}{P}\left[\frac{\sum_{s_1\in\mathcal{G}} z(\boldsymbol{x}_{s_1})^T}{\left\|\sum_{s_1\in\mathcal{G}} z(\boldsymbol{x}_{s_1})\right\|_2}, \cdots, \frac{\sum_{s_P\in\mathcal{G}} z(\boldsymbol{x}_{s_P})^T}{\left\|\sum_{s_P\in\mathcal{G}} z(\boldsymbol{x}_{s_P})\right\|_2}\right]^T. \tag{2.11}$$

In our case, the L_2 normalization strategy has several advantages: i) the overall kernel value is in $(0, 1]$ with the kernel $K(\mathcal{G}, \mathcal{G}) = 1$; ii) the kernel value of each length p contributes equally to the overall kernel value; iii) the normalization strategy maintains the vector form of the set of subpaths, which is suitable for large-scale classification tasks based on linear machine learning algorithms. Note that the inner product $\tau(\boldsymbol{s})^T\tau(\boldsymbol{s}')$ is a valid kernel as it is equivalent to a sum of kernels computed on each length p then divided by P^2.

Further, in SBoSK, we propose to use the maximum considered subpath lengths in Eq. (2.11) instead of using all lengths. This leads to a smaller vector size to be fed into machine learning algorithms, and further reduces the computational time as smaller patterns are needed to be considered.

2.4.4 Algorithm and complexity

The proposed approximation, SBoSK, defined in Algorithm 1 yields a linear complexity of $O(n\ |\mathcal{G}|\ dD)$, while the exact computation maintains a quadratic complexity of $O(n^2\ |\mathcal{G}|\ d)$. Fig. 2.2 illustrates BoSK computed on a pair of trees $\mathcal{T}, \mathcal{T}'$ and its scalable SBoSK extension.

The advantage of the RFF embedding can be easily derived from here: i) instead of pairwise kernel computing with $O(|\mathcal{G}|^2\ d)$, the proposed algorithm computes RFF embedding in $O(|\mathcal{G}|\ dD)$, which is linear *w.r.t.* the structure size $|\mathcal{G}|$, thus allowing the use of the proposed structured kernel in real world application of large structure size; ii) the embedded vector can be fed into a linear machine for training (see in Algorithm. 2 with linear SVM previously presented in Sec. 1.3.2), which yields a linear dependence *w.r.t.* size of training samples of $O(n)$, instead of a non-linear kernel machine that needs to compute a complete kernel matrix with a quadratic complexity of $O(n^2)$.

Algorithm 1: SBoSK embedding using Random Fourier Features

Input: a structured data instance $\mathcal{G}$. *i.e.* a path $\mathcal{P}$ or a tree $\mathcal{T}$

Extract subpaths s_p for length $p = 1 : P$ from $\mathcal{G}$

for *each length* $p = 1 : P$ **do**

  Draw the weights $\boldsymbol{\omega}$ from Gaussian distribution $\boldsymbol{\omega} \in \mathbb{R}^{pd \times D/2}$

  for *each subpath* s_p **do**

    Construct subpath features $\boldsymbol{x}_{s_p}$ as the concatenation of the features of each node

    Compute $z(\boldsymbol{x}_{s_p})$ according to Eq. (2.9)

  end

  Compute $\dfrac{\sum_{s_p \in \mathcal{G}} z(\boldsymbol{x}_{s_p})^T}{\left\| \sum_{s_p \in \mathcal{G}} z(\boldsymbol{x}_{s_p}) \right\|_2}$

end

Output: embedded vector $\tau(\boldsymbol{s})$ as defined in Eq. (2.11)

Algorithm 2: SBoSK with linear SVM for training and testing

Input: training data set $\{\mathcal{G}_i^{\text{train}}\}$, training labels $\{Y_i^{\text{train}}\}$ and testing data set $\{\mathcal{G}_j^{\text{test}}\}$

Training phase:

for *each training data instance* $\mathcal{G}_i^{train}$ **do**

  Compute embedded vector τ_i^{train} using Algorithm. 1

end

Train a linear SVM using $\{\tau_i^{\text{train}}\}$ and $\{Y_i^{\text{train}}\}$

Prediction phase:

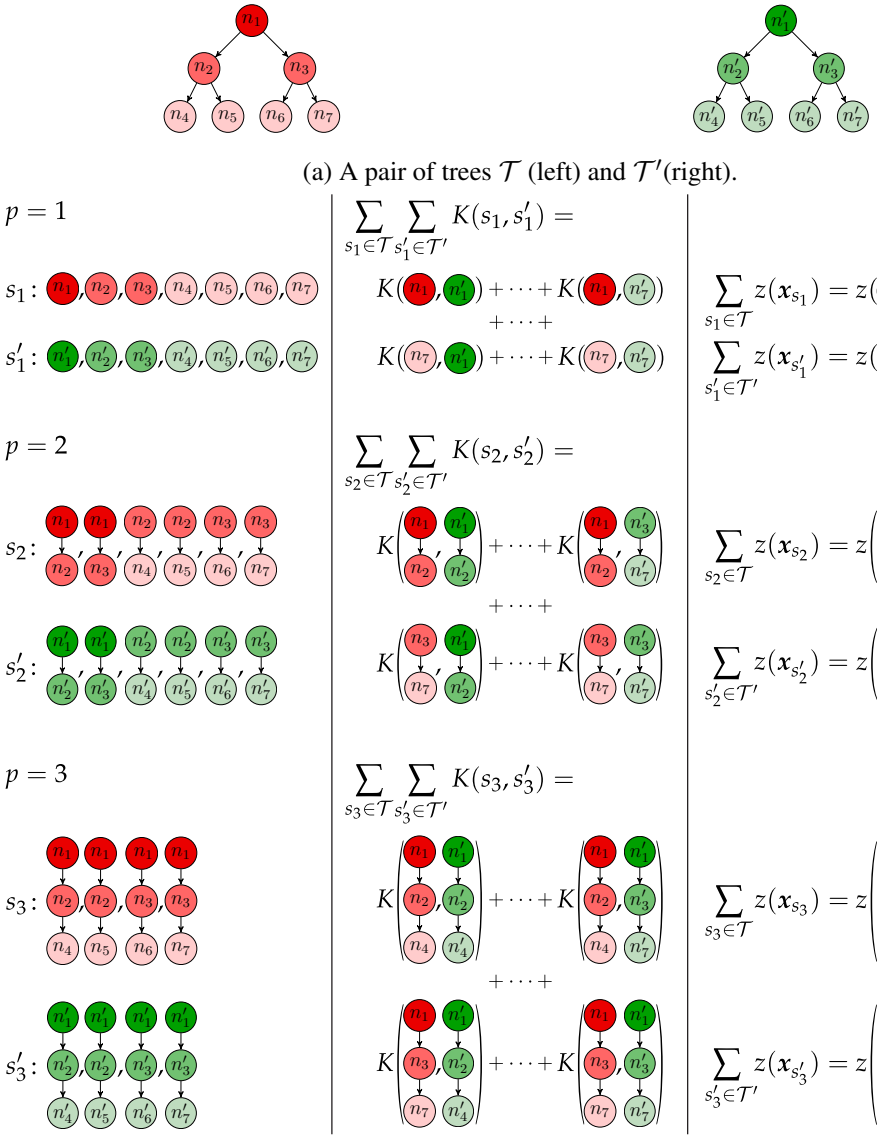
for *each testing data instance* $\mathcal{G}_j^{test}$ **do**

  Compute embedded vector τ_j^{test} using Algorithm. 1

  Predict the label Y_j^{test} using τ_j^{test} and the learned SVM model

end

Output: Predicted labels $\{Y_j^{\text{test}}\}$



(a) A pair of trees $\mathcal{T}$ (left) and $\mathcal{T}'$(right).

(b) subpaths for $\mathcal{T}, \mathcal{T}'$

(c) BoSK

(d) SBoSK

Figure 2.2: Illustration of a pair of trees $\mathcal{T}$ and $\mathcal{T}'$ (Fig. 2.2a) with their subpaths s_p, s'_p (Fig. 2.2b), the computation of BoSK (Fig. 2.2c according to Eq.(4.1)) and SBoSK (Fig. 2.2d according to Eq.(2.10) and Eq.(2.11)). BoSK requires the computation of pairwise kernel value for all training samples, yielding a quadratic complexity *w.r.t.* structure size $O(|\mathcal{G}||\mathcal{G}'|)$ and training sample size $O(n^2)$, while SBoSK only needs the computation of the RFF embedded vector $\tau(\boldsymbol{s})$ for each structure, yielding a linear complexity *w.r.t.* structure size $O(|\mathcal{G}| + |\mathcal{G}|')$ and training sample size $O(n)$.

2.5 Conclusion

In this chapter, we developed a structured kernel dedicated to unordered tree and path (sequence of nodes) structures equipped with numerical features, called Bag of Subpaths Kernel (BoSK). The kernel is an instance of a convolution kernel defined on a complex structure by summing up kernels computed on its substructures. BoSK considers the subpaths as relevant substructures, that is to say — a bag of all paths and single nodes, allowing capturing the vertical hierarchical relationships among nodes in the structured data. The direct computation of BoSK can be done with an iterative scheme, yielding a quadratic complexity *w.r.t.* structure size (number of nodes), and *w.r.t.* volumes of data (training size). Such computation is efficient for small structures and small training data size.

For large-scale problems where the structure can have hundreds of nodes and the available training data can be dozens of thousands or more, we proposed a scalable version of the algorithm (called Scalable BoSK – SBoSK for short) using Random Fourier Features technique. Such a technique maps the structured data in a randomized finite-dimensional Euclidean space, where inner product of the transformed feature vector approximates BoSK. It brings down the complexity from quadratic to linear *w.r.t.* structure size and *w.r.t.* volumes of data, making the kernel compliant with the large-scale machine learning context.

In the following chapters, we consider different applications of (S)BoSK on remote sensing image classification problems under various scenarios.

Chapter 3

Multiscale context-based pixel-wise classification

Contents

In the previous chapter, we presented, from a methodological point of view, the structured kernel (S)BoSK that can be applied for tree and path structures with nodes equipped with numerical features. In the following chapters, we introduce its applications for incorporating the different types of topological information across the scales from a hierarchical representation.

We begin this chapter with the first application of (S)BoSK on path structure, allowing us to take into account the spatial context of a pixel (leaf of the hierarchical representation) through its ancestral regions at multiple scales.

3.1 Introduction

Pixel-wise image classification is a popular topic in the computer vision and in the remote sensing community [29, 15, 139]. The objective of such a classification task is to associate to each pixel in the image one label from a list of predefined classes.

In the standard way to perform pixel-wise classification, each pixel is represented by its spectral information, for instance r-g-b color information, or hyperspectral information in the hyperspectral remote sensing imagery. The spectral feature can be written as a d-dimensional vector and fed directly into a classifier [133]. In such way, each pixel is treated independently, thus the spatial relationships among them are not preserved. However, unlike conventional assumption in machine learning technique where data instances are independent and identically distributed random variables, neighboring pixels have strong correlations [62, 76]. Spatially closed pixels often share similar spectral information and are more likely to belong to the same class. Without taking this image domain specification into account in the classification scheme, the resulting classification maps are often noisy and suffer from the "salt-and-pepper" effect [62].

In order to consider this specific aspect of images, integrating the contextual information has been identified as one of the key solutions for pixel-based classification systems [25, 157, 62, 76]. This information is often revealed through the neighborhood of each pixel, *e.g.* median spectral value within a region generated by morphological area filtering [61]. Neighborhoods are often defined at multiple levels through a hierarchical representation for providing richer information [25, 76, 113]. Spatial context of a pixel at bottom scale (leaf of the hierarchical representation) can be modeled by its ancestral regions at multiple scales as in Fig. 3.1. The multiscale contextual information helps disambiguating similar regions during the classification phase [25]. For instance, individual tree species at the bottom scale can be classified into residential area instead of forest zone given surrounding regions (extracted from a coarser scale in the hierarchical representation) being buildings and roads. Integrating such information leads to classification accuracy improvement and produces spatially smoother classification maps [25, 113].

The hierarchy from a pixel to the whole image can be modeled by a path structure, where the nodes encode the feature of the regions and the edges model the hierarchical relationships among them. SBoSK applied on path structure allows explicitly taking into account the hierarchical relationships among ancestral regions from different scales, providing a powerful tools for multiscale context-based pixel-wise image classification.

The chapter is organized as follows: a brief review of related work considering contextual information is provided in Sec. 4.2. In Sec. 4.3, we describe the proposed pixel-wise classification approach using (S)BoSK. Then we provide a detailed analysis on a synthetic dataset in Sec. 4.4, followed by evaluations on a multispectral remote sensing image in Sec. 3.5, on

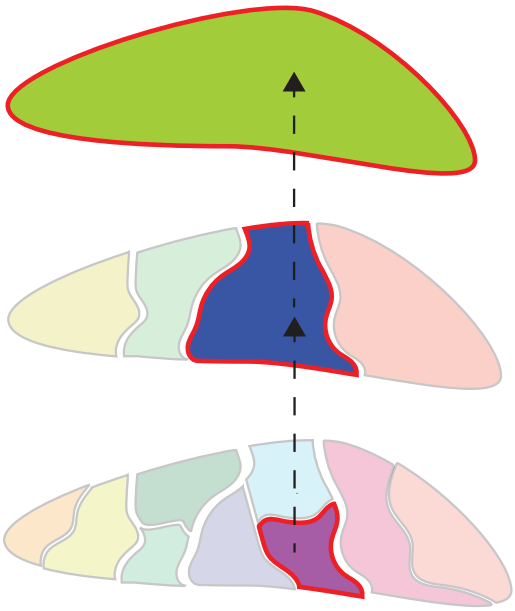


Figure 3.1: Illustration of a hierarchy of objects generated from hierarchical image representation, where the spatial context of the region at bottom scale can be defined through its ancestral regions.

various hyperspectral datasets in Sec. 3.6, and on a large-scale remote sensing dataset in Sec. 3.7.The last section is devoted to conclusion and discussion.

3.2 Related work

In the literature, various methods have been proposed for integrating the spatial contextual information. In this section, we briefly review two main research directions: spatial regularization and spatial feature extraction.

Spatial regularization

Spatial regularization has been successfully applied for guiding the neighboring pixels to produce the same class label for improving the classification accuracy and smoothing the classification map. For instance, pixels in the same region (obtained with image segmentation) are associated to the dominant class label within the region [183], or in [197] neighboring pixels regularization is defined on a hierarchical representation.

In the remote sensing community, especially within the GEOBIA framework [17], incorporating contextual information is commonly achieved through constructing regularization rule-sets for classifying objects and refining classification results. For instance, in [121], the

authors propose to construct rule-sets to classify objects such as roads and vehicles, and to refine the classification results using the spatial relations between classified objects: such contextual information can help distinguishing the moving vehicles from the ones in the parking lots, as they are often surrounded by a road. In [5, 159], spatial relationships among objects are also taken into account with rule-sets. Although designing the knowledge-based rule-set is straightforward to integrate contextual information into classification, it often requires human involvement and interpretation, which is subjective and hard to adapt to new locations and datasets.

A common way to automatically achieve spatial regularization is through Conditional Random Fields technique (CRF) [146]. The CRF defines an energy model containing two terms: the unary potential that measures likelihood of an object belonging to certain classes based on its appearance, and the pairwise potentials that model the pairwise relationships between objects. However, training of a complex model to encode the interaction among classes as in [204] is extremely costly, and it often requires manual annotation of full scenes. Therefore, most of remote sensing applications use a simplified version of CRF only to penalize the neighboring pixels being classified into different classes, thus enforcing smoothness over adjacent regions and increasing the classification accuracy (as known as Potts model) [53, 165, 184]. But the regularization parameters inside these models need to be carefully tuned to avoid under or over smoothness [138, 165], and such parameters are often set empirically and manually by checking the produced classification maps.

Spatial feature extraction

Another interesting research direction for incorporating contextual information is done through spatial feature extraction step. Spatial features can be extracted at the image region level obtained by image segmentation techniques, while the spectral features are extracted at the conventional pixel level. In the end, both spatial and spectral features are combined together and fed into the classifier. For instance, in [61], spatial features are computed as the median spectral information of the segmented region that covers the pixel.

Extracting the contextual information can also be done with hierarchical representation of images [25]. Through the hierarchy, contextual information can be modeled with the features of its ancestral regions at different scales. One of the most popular examples is the Attribute Profile relying on Max- and Min- tree [48], or Tree of Shapes [30]. Integrating contextual information leads to classification accuracy improvement *w.r.t.* using only spectral information [113, 96]. Since the spatial position is also implicitly taken into account, it often produces a spatially smoother classification map avoiding "salt and pepper" effect [25, 96, 113].

Conventional methods such as Attribute Profile [76] or other multiscale features [25, 96,

113] concatenate the features of each ancestral region into a unique (long) stacked vector, before feeding them into a classifier. Therefore, the stacked vector is a set of highly dimensional features. Consequently, as mentioned in [76], it should be properly handled in order to a make full exploitation of the discriminative information, due to the issues raised by the very large dimensionality (Hughes phenomenon) and high redundancy [152].

The hierarchy of objects from a pixel to the whole image can be modeled by a path structure. In the next section, we introduce our proposed (S)BoSK on such path structures for better exploiting the contextual information. We also illustrate the similarities and differences between (S)BoSK and kernel built on stacked vector.

3.3 (S)BoSK on path for multiscale contextual information

We illustrate how (S)BoSK can be applied on ancestral regions that model the spatial context of each pixel through hierarchical image representation. In this case, each pixel is considered as the elementary unit to be classified. It is represented as the leaf of the tree and is described by features of the set of regions linking it to the root. We call this structured type of data a path $\mathcal{P}$, or a sequence of nodes. As illustrated in Fig. 3.2, through the hierarchy, the ancestral regions encode the evolution of pixel from finer to coarser level, thus contextual information can be revealed.

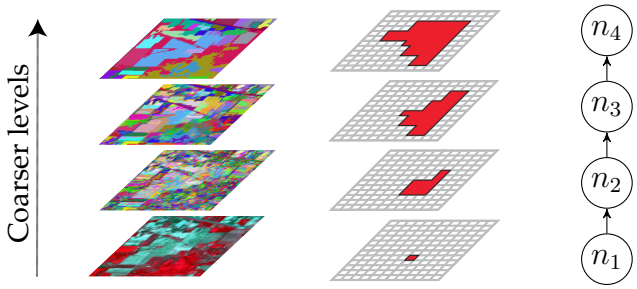


Figure 3.2: Contextual information extracted from a hierarchical image representation. Each pixel (leaf of the hierarchical representation) is considered as a data instance to be classified, and is described by features extracted from the set of ancestral regions on the path $\mathcal{P}$ linking it to the root.

Formally speaking, let n_1 be a pixel of the image. Through a hierarchical image representation, we write n_i the nested image regions at level $i = 2, ..., P$, with region at lower levels always being included in higher levels *i.e.* $n_1 \subseteq n_2 \ldots \subseteq n_P$. The contextual information of a pixel n_1 can then be described by its ancestral regions n_i at multiple levels $i = 2, ..., P$. More specifically, one can define the contextual information as a path or sequence $\mathcal{P} = (n_1, ..., n_P)$ that encodes the evolution of the pixel n_1 through the different levels of the hierarchy. Each node n_i is described by a d-dimensional feature $\boldsymbol{x}_{n_i}$ that encodes the region characteristics

e.g. spectral information, size, shape, etc.

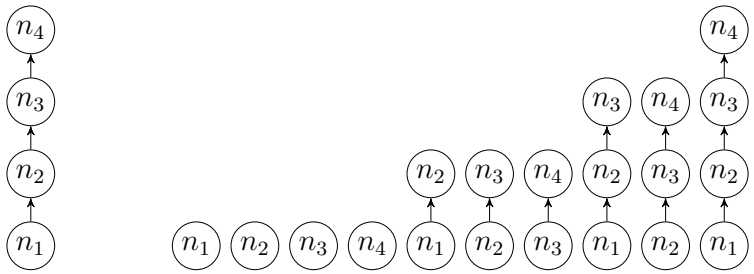


Figure 3.3: A path $\mathcal{P}$ and its subpaths s_p.

In order to capture the vertical hierarchical relationships between the nodes, we decompose the path $\mathcal{P}$ as a set of subpath substructures. A subpath can be defined as contiguous subsequences on a path, as illustrated in Fig. 3.3. BoSK between $\mathcal{P}$ and $\mathcal{P}'$ can be computed as:

$$K_{BoSK}(\mathcal{P}, \mathcal{P}') = \sum_{p=1}^{P} \mu_p \sum_{s_p \in \mathcal{P}} \sum_{s'_p \in \mathcal{P}'} K(s_p, s'_p) \ , \tag{3.1}$$

where μ_p is a weighting parameter allowing one to vary the contributions of overall kernel value of each subpath lengths (see Sec.2.3.3), and $K(s_p, s'_p)$ is the kernel between two subpaths s_p and s'_p, which is defined as the product of atomic kernels computed on pairs of nodes $k(n_{(t)}, n'_{(t)})$ of the subpaths, following an ascending order $1 \leq t \leq p$:

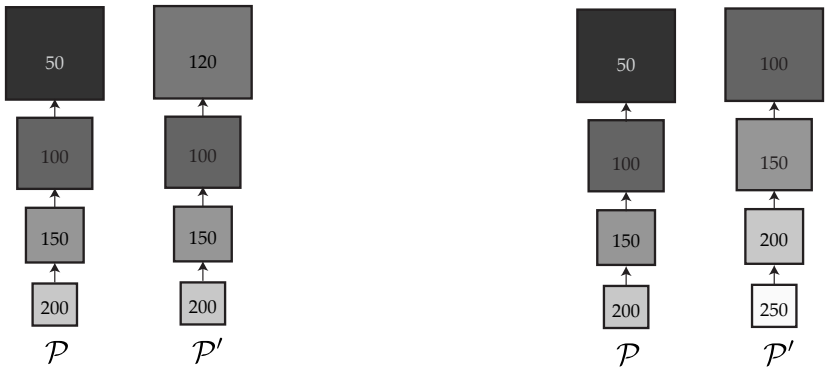
$$K(s_p, s'_p) = \prod_{t=1}^{p} k(n_{(t)}, n'_{(t)}) \ . \tag{3.2}$$

We can see here that the kernel computed on the stacked vector *i.e.* concatenation of the nodes features (denoted as $K_{SV}(\cdot)$) is actually a special case of (S)BoSK when using Gaussian kernel for the atomic kernel:

$$\begin{aligned} K_{SV}(\mathcal{P}, \mathcal{P}') &= \exp(-\gamma \|H_{\mathcal{P}} - H_{\mathcal{P}'}\|^2) = \exp\left(\sum_{i=1}^{P} (-\gamma \|\boldsymbol{x}_{n_i} - \boldsymbol{x}_{n'_i}\|^2)\right) \\ &= \prod_{i=1}^{P} \exp(-\gamma \|\boldsymbol{x}_{n_i} - \boldsymbol{x}_{n'_i}\|^2) = K(s_P, s'_P) \ , \end{aligned} \tag{3.3}$$

where $H_{\mathcal{P}} = [x_{n_1}, x_{n_2}, \cdots, x_{n_P}]$ is the concatenation of the nodes features $\boldsymbol{x}_{n_i}$. When using the Gaussian kernel with $H_{\mathcal{P}}$, it corresponds to BoSK computed on the subpath with maximum length s_P only, with $|s_P| = |\mathcal{P}|$.

Let us recall that the stacked vector is commonly used for multiscale context feature representation in remote sensing, representative examples of this framework including attribute



(a) Two similar paths with a difference only at the top level.

(b) Two similar paths with common structures across the levels.

Figure 3.4: Example of similar paths $\mathcal{P}, \mathcal{P}'$ with a certain level transformation.

profiles [76], as well as other multiscale features [25, 113, 96]. However, we argue that, compared to the kernel built on stacked vector, BoSK can better take into account the specific nature of the data generated from hierarchical image representation. We illustrate such a superiority as follows with two straightforward examples.

We construct two paths $\mathcal{P}$ and $\mathcal{P}'$ extracted from a hierarchical representation with region feature been defined as gray level (in Fig. 3.4). To compare K_{BoSK} and Gaussian kernel on stacked vector K_{SV}, we use the atomic kernel in K_{BoSK} and K_{SV} with high gamma value *i.e.* $\gamma = 10000$ for the sake of illustration, which ensures that kernel value equals to 1 when two region have the same gray level, otherwise close to 0. Note that we have self-similarity equals to 1 for both kernels: $K_{SV}(\mathcal{P}, \mathcal{P}) = 1$, $K_{BoSK}(\mathcal{P}, \mathcal{P}) = 1$, (*resp.* $\mathcal{P}'$).

In the first example (as in Fig. 3.4a), $\mathcal{P}$ and $\mathcal{P}'$ are the same except at the last level. This is a common situation using multi-scale segmentation algorithms on real world images, where regions at coarser levels of the hierarchy are large and often consists of several classes, thus are more likely to be inconsistent. We have $K_{BoSK}(\mathcal{P}, \mathcal{P}') = 0.6$, as $\mathcal{P}$ and $\mathcal{P}'$ share most of the parts. However, due to the different nodes at last level, we have $K_{SV}(\mathcal{P}, \mathcal{P}') = 0$, indicating that the two paths are completely different.

In the second example (as in Fig. 3.4b), $\mathcal{P}$ and $\mathcal{P}'$ are similar as they share certain cross-scale patterns. This might happen when objects-of-interest lay at different scales. We have $K_{BoSK}(\mathcal{P}, \mathcal{P}') = 0.6$, as $\mathcal{P}$ and $\mathcal{P}'$ share majority of the parts across different scales. However, for Gaussian kernel on stacked vector, we have $K_{SV}(\mathcal{P}, \mathcal{P}') = 0$. Again it indicates that the two paths are completely different.

3.4 Experiments on a synthetic dataset

We study here the behavior of the proposed (S)BoSK on path structures though the following scenario using an artificial dataset. In Fig. 3.5, two classes consist of similar types of leaves (data instances to be classified) at bottom level (type A and B), which can not be distinguished using only bottom scale. Such a scenario can be found in the pixel-wise remote sensing classification context, where pixels with different spectral information might be defined as the same class, corresponding to the intra-class diversity, or pixels sharing the similar spectral information might belong to different classes, referring as inter-class correlation. However, due to the different spatial arrangement of image contents, it might generate new different nodes at intermediary level, also affecting vertical relationships between these nodes. Therefore, the evolution of pixels in the hierarchy changes between different classes, providing discriminative contextual information.

3.4.1 Dataset description and experimental setup

We use two types of leaves, A and B, that are described by a 1-D feature generated according to a uniform distribution, with non overlapping intervals, $A \sim U(0,5)$ and $B \sim U(5,10)$. Number of leaves and node merging parameters are defined randomly to produce various shapes of trees within each class. In our evaluation, we have for each tree about 400 leaves and the maximum depth is 15. The features of each node are average and variance features of the leaves that compose the nodes. We then generate two classes as shown in Fig. 3.5 with different multiscale merging configurations by forcing type A leaves to merge with type B leaves in `Class 1`, while in `Class 2`, type A (*resp.* B) leaves always merge with type A (*resp.* B).

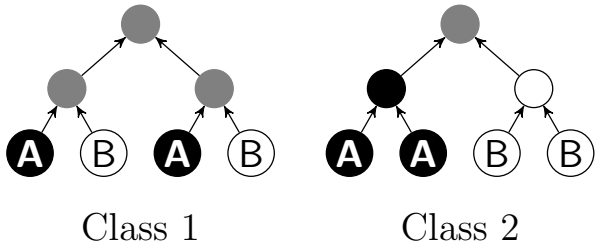


Figure 3.5: Synthetic concept for experimental evaluations. Each leaf of the tree is considered as a data instance to be classified, and is described by features extracted from the set of regions on the path linking it to the root.

We consider a one-against-one SVM classifier (using the Python implementation of LibSVM [31]) with the Gaussian kernel as the atomic kernel. All free parameters are determined by five-fold cross-validation, which include: the bandwidth γ of Gaussian kernel and the

SVM regularization parameter C over potential values.

The Gaussian kernel on stacked vector and BoSK with different weighting strategies are compared. We considered three weighting schemes with different μ_p in Eq. (3.1): i) BoSK with constant weights; ii) limitation of the maximum length of substructures [39]; iii) the use of exponential weighting [101, 39]. BoSK weighting parameters are determined by five-fold cross-validation: the exponential parameter $\lambda \in (0,1)$ and the maximum considered sub-path length $P \in \{1,2,\dots 15\}$ with 15 being the maximum length of the path in the dataset.

Accuracies (and standard deviations) of each setup are computed after 10 repetitions of each experiment, choosing randomly 100 data instances of each class as training samples and another set of 100 data instances for testing.

We observe in Tab. 3.1 that the use of a Gaussian kernel with only leaves attribute at bottom scale provides an accuracy about 50%, because of the non discriminative leaves of the two classes. The contextual information, especially the one revealed by discriminative nodes at intermediary levels, can be easily captured by the Gaussian kernel on stacked vector and BoSK, leading to a 100% accuracy.

Table 3.1: Mean (and standard deviation) of overall accuracies (OA) computed over 10 repetitions.

Method	**OA**[%]
Gaussian kernel on leaf	50.0 (2.8)
Gaussian kernel on stacked vector	100.0 (0.0)
BoSK on path	100.0 (0.0)

3.4.2 Overall evaluation of BoSK and Gaussian kernel on stacked vector

We study the behavior of BoSK and compare it with the Gaussian kernel computed on the stacked vector by adding some confusion or noise inside the two classes. Two particular behaviors have been studied as follows:

(1) Robustness to outliers. We modify the scenario to introduce outliers at bottom scale that take values outside ranges of type A and type B leaves. In our experiments, we choose outliers $\sim U(10,30)$. The ratio of such leaves varies from 0% to 100%.

(2) Robustness to mislabeled leaves. We introduce some mislabeled leaves in the trees. To do so, a given percentage of leaves of type A (randomly chosen) are changed into type B, and vice versa. In the binary classification setup considered here, the ratio of mislabeled leaves in each class varies from 0 % to 50 %, leading to a more confusing structure between the two classes.

Fig. 3.6 and 3.7 present the accuracies obtained in these two settings. We notice that in both scenarios, BoSK maintains a good performance up to a certain ratio of structure distor-

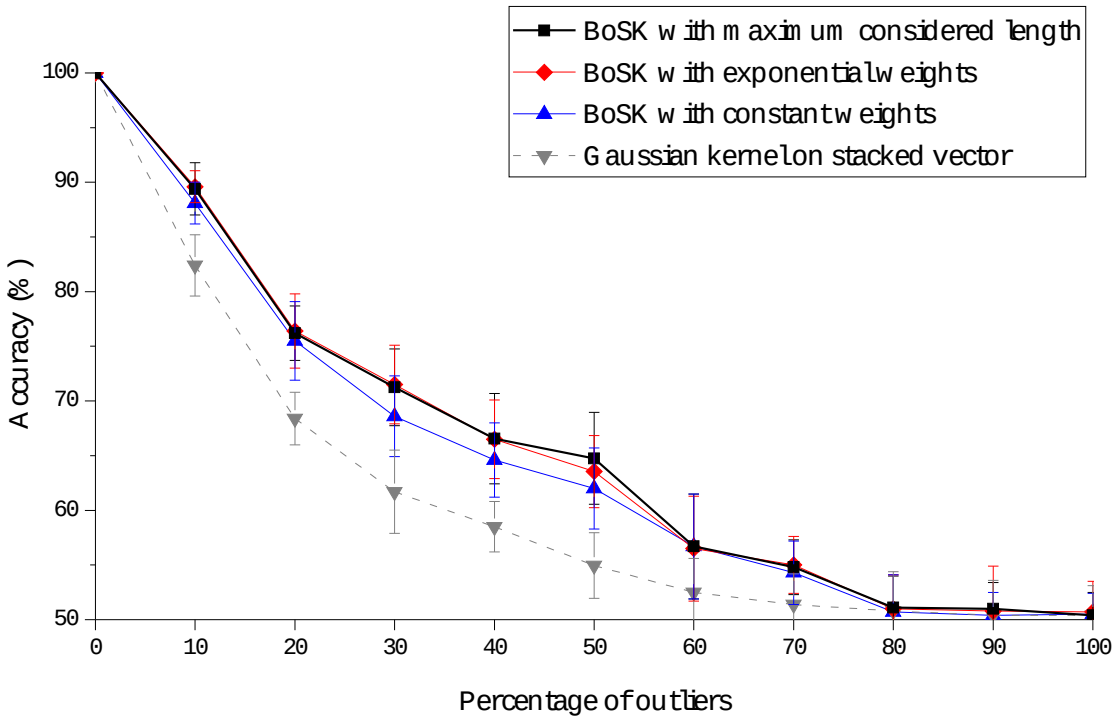


Figure 3.6: Accuracies and standard deviations of BoSK with various weighting schemes and Gaussian kernel on stacked vector in presence of outliers.

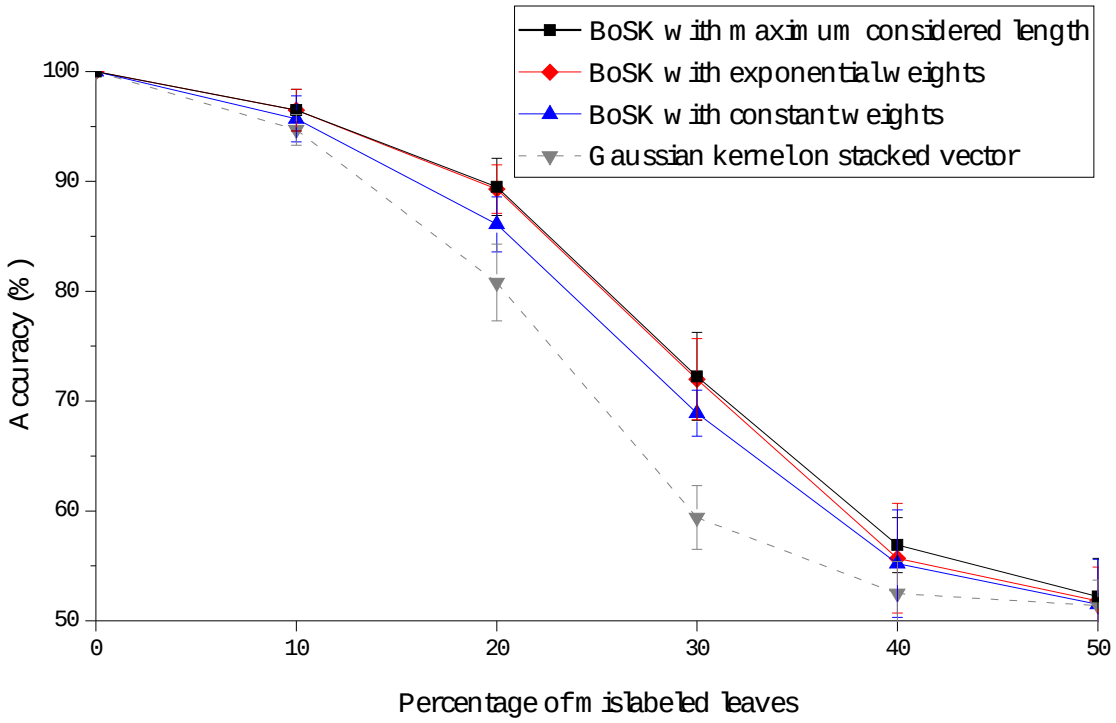


Figure 3.7: Accuracies and standard deviations of BoSK with various weighting schemes and Gaussian kernel on stacked vector in presence of mislabeled leaves.

tion (with all three weighting strategies), yielding consistent improvements over the Gaussian kernel on stacked vector. In addition, we can observe that using different weighting strategies, such as exponential weighting and use of maximum length of subpaths considered, can affect the results and the later yields slightly better results in both scenarios.

Impact of the maximum considered subpath length

We now analyze the impact of maximum subpath length P of BoSK and illustrate the reasons why BoSK can obtain better results compared to the Gaussian kernel on stacked vector. Note that the exponential weighting and maximum considered subpath length work in a similar way. They both aim at limiting the impact of subpaths with longer lengths, while analyzing the impact of the maximum considered subpath length is more straightforward.

To do so, we compute the accuracies with different maximum subpath lengths $P \in \{1, 2, \ldots, 15\}$ using two scenarios *i.e.* 30% of outliers and 30% of mislabeled leaves, for ease of analysis.

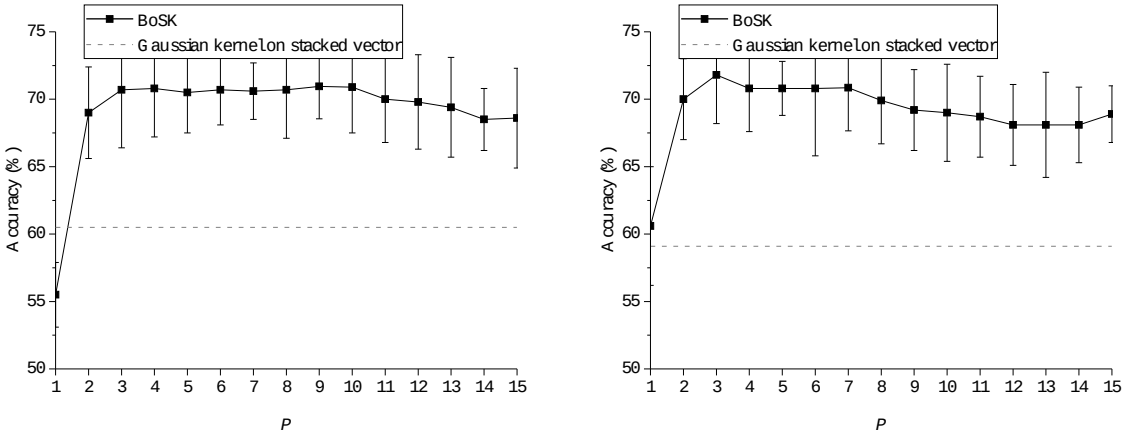
Fig. 3.8 shows that the maximal accuracies are obtained with $P = 3$, where we see a clear accuracy gain compared to the Gaussian kernel on stacked vector. In addition, we observe that the accuracies increase greatly compared to BoSK built on nodes only *i.e.* $P = 1$, then gradually decrease when considering longer subpaths into BoSK computation.

The decrease of performance when adding longer subpath calls for a deeper analysis of the performance on each subpath length individually. To do so, we show the accuracies on individual subpath length $p \in \{1, 2, \ldots, 15\}$. As we can observe in Fig. 3.9, the performances are lower when p is large. Therefore, the classification accuracies of BoSK might drop down when many less discriminative kernels are added [26]. This is the main reason for which BoSK built with maximum considered subpath length performs better than BoSK with constant weights. Such observation calls for penalization of longer subpath patterns.

In addition, we observe in Fig. 3.9 that BoSK built on only subpaths with maximum length $p = 15$ yields one of the worst results among other individual subpath lengths. In fact, this corresponds to the Gaussian kernel on stacked vector.

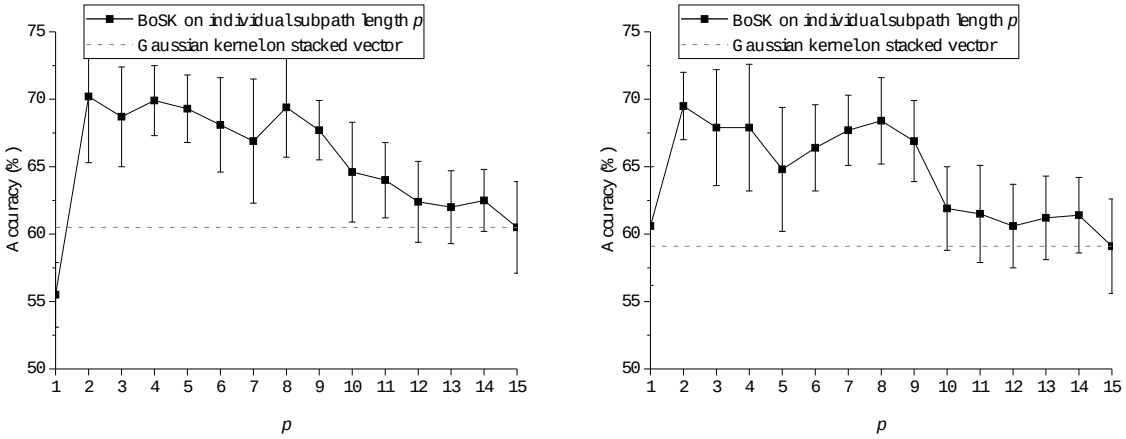
3.4.3 SBoSK analysis

In this section, we study the behavior of the scalable version of BoSK, called SBoSK. Following the same configuration of class generation, we analyze SBoSK through the scenario of 30% mislabeled leaves, as this setting can lead to a reasonable confusion between the two classes for the sake of analysis.



(a) in presence of outliers (b) in presence of mislabeled leaves

Figure 3.8: Classification accuracies of BoSK using maximum considered subpath length P and Gaussian kernel on stacked vector in presence of outliers and mislabeled leaves.



(a) in presence of outliers (b) in presence of mislabeled leaves

Figure 3.9: Classification accuracies of BoSK using individual subpath length p and Gaussian kernel on stacked vector in presence of outliers and mislabeled leaves.

Kernel approximation analysis

In order to analyze the quality of the approximated structured kernel SBoSK *w.r.t.* the dimension of the Random Fourier features D in Eq. (2.11), we use the matrix approximation error computed between the approximation matrix $\check{K}$ and the exact kernel matrix K by matrix Frobenius norm, as used in [210]:

$$error = \frac{\left\|K - \check{K}\right\|_F}{\|K\|_F} \tag{3.4}$$

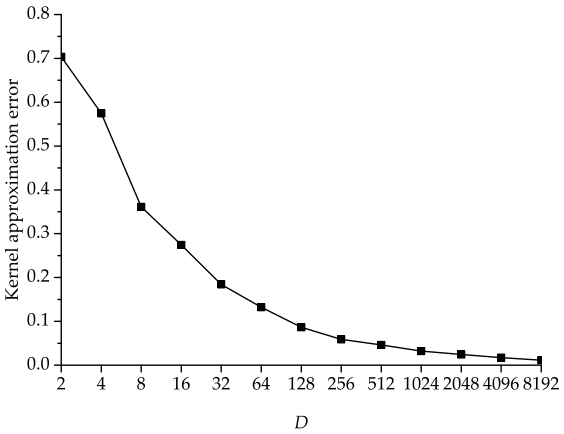


Figure 3.10: SBoSK approximation error *w.r.t.* Random Fourier Features dimension D (log scale).

Kernel normalization strategy is used here for analysis of approximation errors, similarly as proposed for BoSK (see Sec. 2.3.3, Eq. (2.3)). In addition, we use the constant weighting scheme for both BoSK and SBoSK with the same γ for atomic Gaussian kernel. In such settings, the results depend only on the quality of Random Fourier Features approximation, without being affected by other hyperparameters. Fig. 3.10 shows the relation between kernel approximation error and Random Fourier Features dimension D. We can observe that the approximation error decreases faster at beginning, then error gradually tends to zero when a larger dimension is considered. This corresponds to the theoretical analysis on the boundary of approximation error in [160, 180].

Classification accuracy

We compare the classification accuracies of BoSK and SBoSK following the same setting as the previous analysis. As we can see in Fig. 3.11, when the Random Fourier Features dimension D increases, the accuracy of SBoSK also increases until it converges to the accuracy obtained by BoSK. This is consistent with previous observation shown in Fig. 3.10: the kernel approximation error tends to zero when a larger dimension D is used. Since kernel matrices obtained by BoSK and SBoSK are similar, both kernels achieve a similar classification accuracy.

Secondly, we considered the L_2 normalization strategy on SBoSK as in Sec. 2.4.3, Eq. 2.11. We observe in Fig. 3.12 that the accuracies improve greatly improve (*i.e.* more than 10%) when using subpaths with various lengths *w.r.t.* using BoSK built using only nodes ($P = 1$). However, accuracies decrease when adding longer subpath patterns, calling for a penalization of these longer subpaths. We thus propose to set a maximum subpath length for SBoSK so that classification accuracy can be kept as high as possible, while leading to a smaller vector size to be fed into machine learning algorithms, which can further reduce the compu-

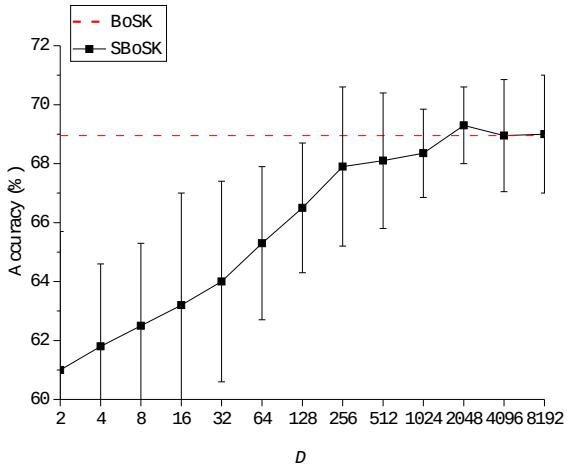


Figure 3.11: Classification accuracy *w.r.t.* Random Fourier Features dimension D (log scale).

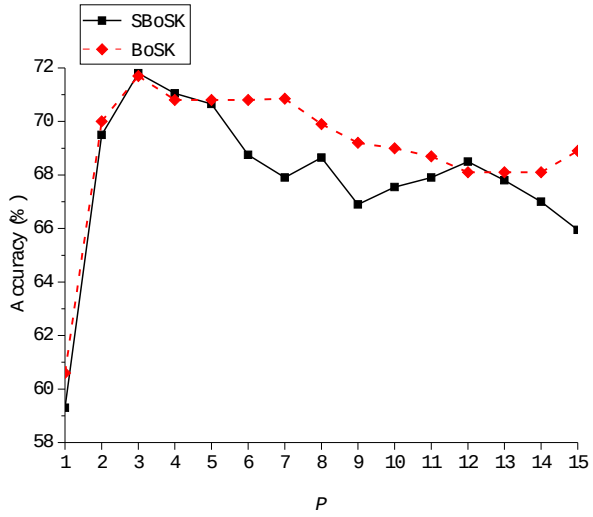


Figure 3.12: Classification accuracy of SBoSK and BoSK *w.r.t.* maximum considered subpath lengths P.

tation time as smaller patterns are being considered.

Note that the L_2 normalization used in SBoSK promotes an equivalent contribution of each subpath length, thus may result in different accuracies obtained by BoSK using maximum considered subpath length. However, such a choice allows benefiting the large-scale classification tasks based on linear machine learning algorithms, as the normalization is done on each data instance individually and as the Random Fourier Features embedding maintains a vector form representation of data.

Complexity analysis

Complexity *w.r.t.* Random Fourier Features dimension

Here we analyze the computation complexity of SBoSK *w.r.t.* Random Fourier Features dimension D. Fig. 3.13a shows that the computation time increases linearly *w.r.t.* dimension $O(D)$. Such a complexity calls for a trade-off between the quality of the kernel approximation and the computational complexity, as the computation time increases linearly with the dimension, while approximation errors decrease slowly when the dimension is large (as shown previously in Fig. 3.10).

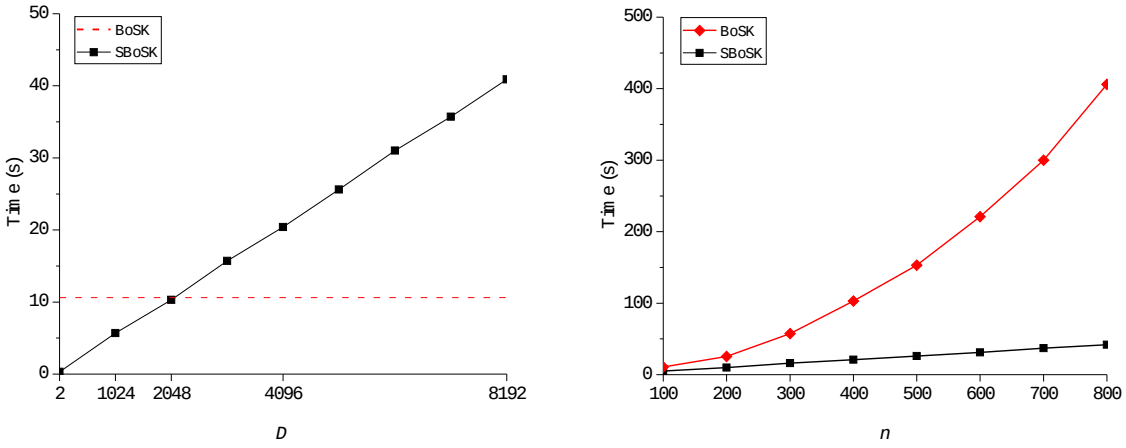


Figure 3.13: Computational time of SBoSK *w.r.t.* Random Fourier Features dimension D (Fig. 3.13a) and *w.r.t.* training samples size n (Fig. 3.13b).

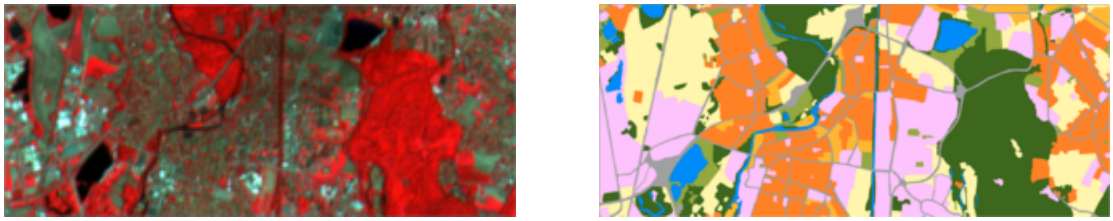
Complexity *w.r.t.* training sample size

We compare here the computation time of BoSK and SBoSK *w.r.t.* training sample size n (we compute here SBoSK with $D = 2048$ for illustration as it yields a similar computational time as BoSK for $n = 100$). Fig. 3.13b shows that the time for BoSK increases quadratically *w.r.t.* training sample size $O(n^2)$, because of the quadratic increase of the number of pairwise BoSK to be computed in the Gram matrix. However, for SBoSK, only a linear increase of $O(n)$ can be observed. The computation of SBoSK is dominated by the Random Fourier Features embedding algorithm, which computes each data instance independently. The Gram matrix can be formed later by computing the inner product of resulting vectors, whose computation time is negligible. Therefore, we observe a linear complexity of $O(n)$.

3.5 Strasbourg Spot-4 image classification

3.5.1 Dataset and design of experiments

In this section, we focus on urban land-use classification in the South of Strasbourg city, France, using Spot-4 satellite image with 20 m resolution. The image is composed of 326 × 135 pixels with 4 spectral bands: Green, Red, NIR (near infrared), MIR (middle infrared). We consider 8 thematic classes of urban patterns as shown in Tab. 3.2 (class details) and in Fig. 3.14b (ground truth image). For more information, see [107] for a detailed description of the dataset.



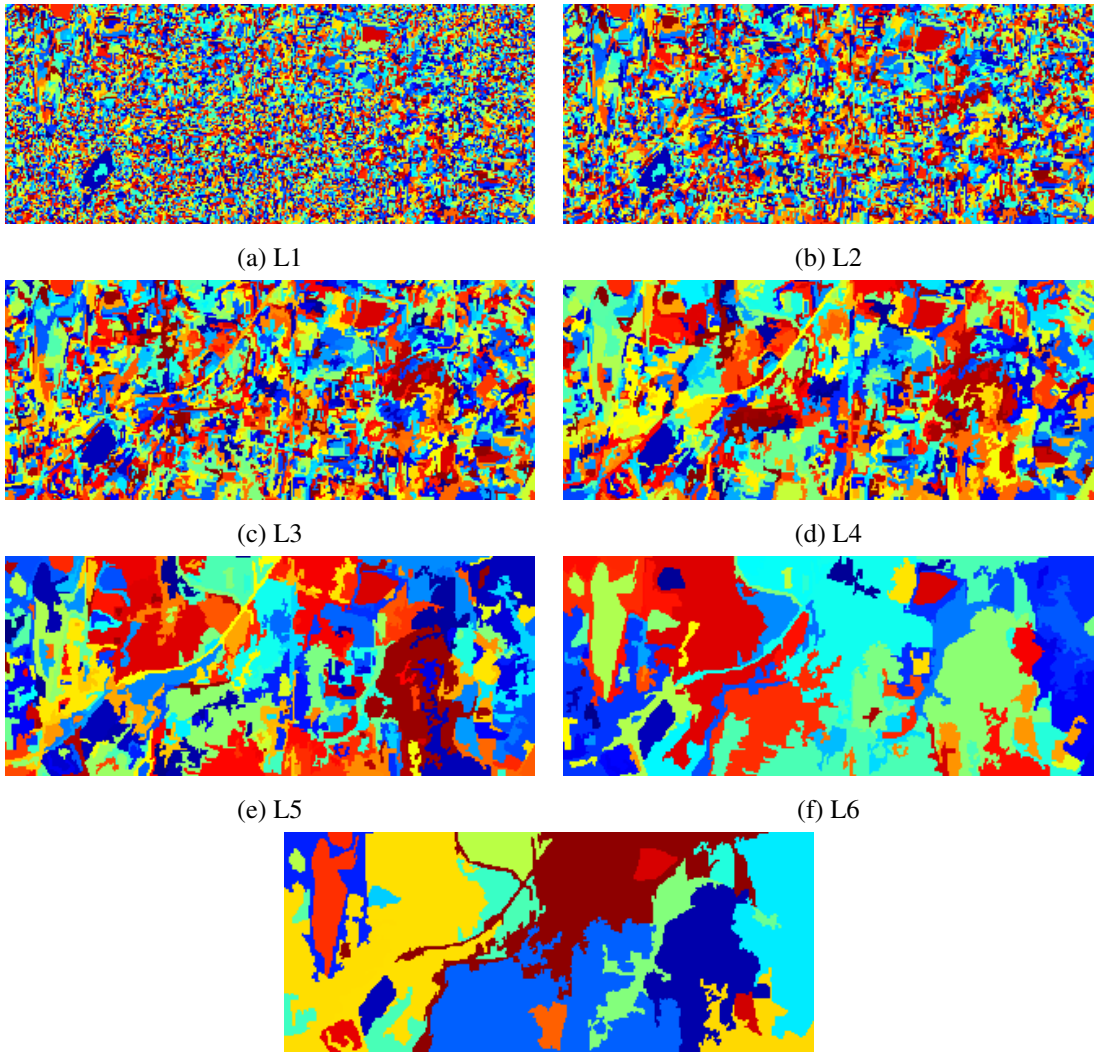
(a) Spot-4 image (b) Ground truth image

Figure 3.14: Urban scene taken over South of Strasbourg, France: (a) false color image of Spot-4 (© CNES 2012) with 20 m resolution and (b) the associated ground truth (© LIVE UMR 7362, adapted from OCSOL CIGAL 2012) with eight thematic classes.

Table 3.2: List of classes, their color, and number of pixels in ground truth in Fig. 3.14b.

Class	Color	Nb of pixels
Water surfaces	Blue	1,653
Forest areas	Dark green	9,315
Urban vegetation	Light green	1,835
Road	Grey	3,498
Industrial blocks	Pink	8,906
Individual housing blocks	Dark orange	9,579
Collective housing blocks	Light orange	1,434
Agricultural zones	Yellow	7,790
Total		**44,010**

Each pixel is considered as one data instance to be classified. In order to extract the contextual information, we generate, from the bottom level consisting of single pixels, 7 additional levels of hierarchical segmentation by increasing the region dissimilarity criterion $\alpha = [2^{-2}, 2^{-1}, ..., 2^{4}]$ (segmentation maps for different scales are shown in Fig. 3.15). We observe that with such parameters, the number of segmented regions is roughly decreasing



(a) L1 (b) L2

(c) L3 (d) L4

(e) L5 (f) L6

(g) L7

Figure 3.15: Hierarchical segmentation maps of Strasbourg Spot-4 dataset. All levels of segmentation are illustrated here by using dissimilarity criterion $\alpha = [2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4]$.

by a factor of 2 between each level.

Each region in the hierarchical representation is described by a 8-dimensional feature vector $\boldsymbol{x}$, which includes the region average of the 4 original multi-spectral bands, Soil Brightness index (BI) and NDVI (the normalized difference vegetation index), as well as two Haralick texture measurements computed with gray level co-occurrence matrix homogeneity and standard deviation. These features are considered as standard ones in the urban analysis context [66].

We consider a one-against-one SVM classifier (using the Python implementation of LibSVM [31]) with Gaussian kernel as the atomic kernel. All free parameters are determined by 5-fold cross-validation, which include: the bandwidth γ of Gaussian kernel and the SVM regularization parameter C over potential values, the maximum considered subpath length $P \in \{1, 2, \ldots, 8\}$. The RFF dimension D is chosen empirically as a trade-off between the computational complexity and the classification accuracy (and will be further analyzed in Sec. 3.5.2). Henceforth, in this section, all reported the results are averaged over 10 repetitions.

3.5.2 SBoSK analysis

In this section, we compare, in terms of classification accuracy and computation time, BoSK and its scalable version SBoSK.

We analyze firstly the impact of the number of RFF dimensions on the accuracies. To do so, we compute BoSK and SBoSK with $D = \{2^1, 2^2, \ldots, 2^{13}\}$ using 400 training samples per class and the rest for testing. As we can observe in Fig. 3.16, when RFF dimension increases, the accuracy increases until it converges to the accuracy obtained with BoSK using the exact computation scheme.

Secondly, we analyze the impact of the RFF dimension in terms of computation time. To do so, we follow the previous setting and use differing training samples per class $n = \{50, 100, \ldots, 1600\}$ (except when $n = 1600$, we use all 1434 available samples for collective housing blocks). As we can see in Fig. 3.17, the computation time increases linearly *w.r.t.* n for SBoSK, while for its exact computation, it increases quadratically. This illustrates the potential of the proposed RFF embedding in SBoSK to be applied in the context of large-scale machine learning. In addition, we observe that the computation time increases linearly *w.r.t.* dimension D, while the accuracy shown in Fig. 3.16 improves only slightly when D is large. Therefore, one might have to find a trade-off between the quality of the approximation and the computation time. Hereafter, we empirically fix the RFF dimension to be $D = 4096$.

In addition, we analyze the impact of the maximum considered subpath length P using the proposed L_2 normalization strategy for SBoSK (Eq.2.11). Fig. 3.18 shows that the accuracies improve when considering subpaths with different lengths *w.r.t.* using only nodes *i.e.*

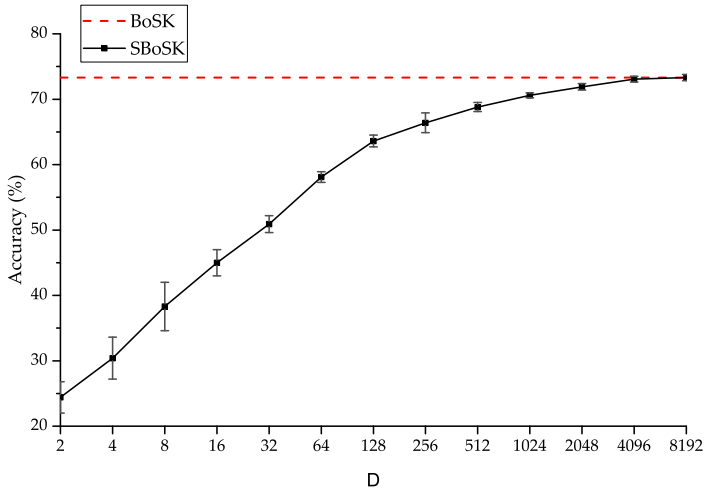


Figure 3.16: Classification accuracy of BoSK and SBoSK with different dimensions D on the Strasbourg Spot-4 image (log scale). Reported accuracies and standard deviation are computed over 10 repetitions with 400 training samples per class.

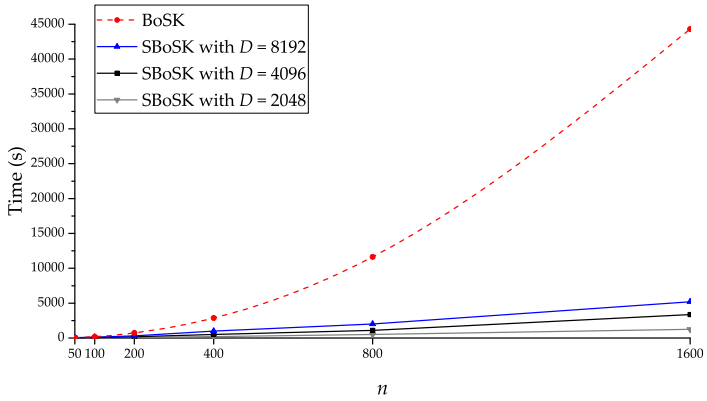


Figure 3.17: Computation time of BoSK and SBoSK with $D = \{2048, 4096, 8192\}$ *w.r.t.* n number of training samples per class.

$P = 1$. However, the accuracies might decrease when adding longer subpaths $P > 5$, thus calling for a penalization of longer subpath patterns. Besides, we propose to set a maximum subpath length for SBoSK, leading to a smaller vector size to be fed into machine learning algorithms, which can further reduce the computation time as smaller patterns are being considered.

3.5.3 Results and analysis

For comparison, we consider the Gaussian kernel computed at the pixel level (without any contextual/spatial information) as the baseline, and compare our work with several well-

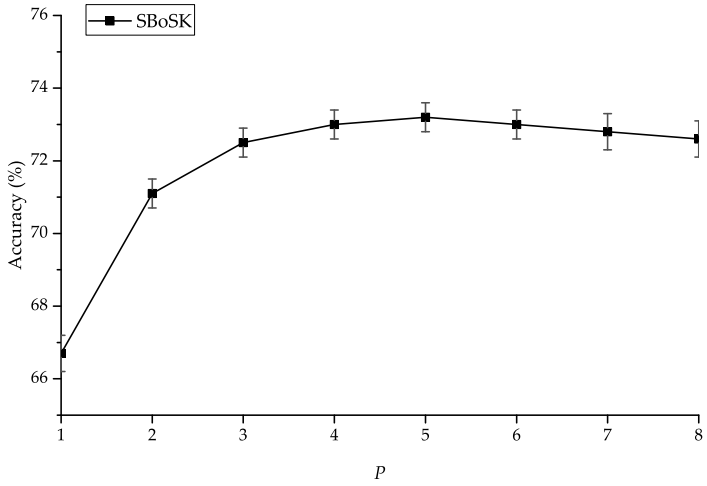


Figure 3.18: Classification accuracy *w.r.t.* different maximum considered subpath lengths P. SBoSK is computed on the Strasbourg Spot-4 image with $D = 4096$.

known techniques for spatial/spectral remote sensing image classification. Spatial-spectral kernel [61] has been introduced to take into account pixel spectral value and spatial information through accessing the nesting region. We thus implement spatial-spectral kernel based on the multiscale segmentation commonly used in this paper, and select the best level (determined by a cross-validation strategy) to extract spatial information. Attribute profile [48] is considered as one of the most powerful techniques to describe image content through context feature. We use full multi-spectral bands with automatic level selection for the area attribute and standard deviation attribute as detailed in [75]. Stacked vector was adopted in [96, 25, 113], and relies on features extracted from a hierarchical representation. We use a Gaussian kernel with stacked vector that concatenates all nodes from ascending paths generated from our multiscale segmentation. The comparison is done by randomly choosing $n = [50, 100, 200, 400]$ samples for training and the rest for testing. The classification accuracies with different methods are shown in Tab. 3.3. Three common accuracy assessment measures in the remote sensing community [15, 40] are reported here: overall accuracy, average accuracy, Kappa statistic. We also give the per-class accuracies using $n = 400$ training samples in Fig. 3.19.

When compared to the Gaussian kernel computed at pixel level using only spectral information, SBoSK can greatly improve the classification accuracies. We observe about 20% consistent accuracy improvement for different training sample sizes. Per-class accuracies indicate that this improvement concentrates on all classes except two: water surface and forest areas for which classification accuracies remain similar since contextual information extracted from ancestral regions through the hierarchy are not very useful in these mostly homogeneous regions.

SBoSK achieves about 5% improvement over spatial-spectral kernel and attribute profiles for various training sample sizes. For these two state-of-the-art methods considering spatial information, the results actually depend on the selected scales. However, for spatial-spectral kernel relying on a single scale, it is hard to define such a single scale that fits all objects, as objects are often revealed through various scales. Therefore, for certain classes, *e.g.* urban vegetation, it might yield good results with the selected scale. However, it is difficult to generalize for all classes. On the other hand, attribute profiles require to set the thresholds for different attributes in order to achieve good classification results. However, as indicated in [76], generic strategies for filter parameters selection for different attributes are still lacking.

Compared to the Gaussian kernel with stacked vector, SBoSK achieves about 8% classification accuracy improvement for various training sample sizes. Since both kernels rely on the same paths, it demonstrates the superiority of SBoSK for taking into account contextual information extracted from a hierarchical representation. In fact, the Gaussian kernel with the stacked vector is actually a special case of BoSK with the subpath length equal to the maximum, However, BoSK built only on the largest subpath are usually not robust. In our experiment, this superiority is presented in the per-class accuracies for all except two homogeneous classes: water surface and forest areas.

Table 3.3: Mean (and standard deviation) of overall accuracies (OA), average accuracies (AA) and Kappa statistics (κ) computed over 10 repetitions for Strasbourg MSR image with different training data size n. Best results (with a statistical significance less than 0.01% *w.r.t.* others considering the Wilcoxon signed-rank test for matched samples) are boldfaced.

n		Pixel	Spatial-spectral	Attribute profile	Stacked vector	SBoSK
50	OA	45.3 (2.3)	53.2 (1.0)	51.9 (2.1)	49.8 (1.8)	**57.8 (1.3)**
	AA	43.9 (1.0)	53.7 (1.4)	51.7 (1.4)	48.4 (1.1)	**57.9 (0.8)**
	κ	32.2 (2.1)	45.1 (1.1)	43.6 (2.4)	41.2 (1.9)	**50.2 (1.4)**
100	OA	47.9 (1.3)	57.7 (0.9)	57.1 (1.4)	54.3 (1.4)	**63.3 (0.7)**
	AA	46.2 (0.5)	59.2 (0.7)	57.3 (0.7)	52.9 (1.0)	**64.0 (0.7)**
	κ	39.1 (1.3)	49.7 (1.0)	49.5 (1.5)	46.3 (1.6)	**56.5 (0.8)**
200	OA	51.4 (0.8)	63.1 (0.9)	61.7 (0.5)	59.0 (0.5)	**68.4 (0.7)**
	AA	48.1 (0.4)	64.6 (0.6)	62.2 (0.2)	57.5 (0.6)	**69.7 (0.5)**
	κ	42.6 (0.8)	56.3 (1.0)	54.7 (0.5)	51.6 (0.5)	**62.3 (0.7)**
400	OA	52.2 (0.4)	67.3 (0.8)	65.0 (0.5)	62.7 (0.6)	**73.0 (0.4)**
	AA	49.1 (0.2)	68.5 (0.5)	66.3 (0.4)	62.6 (0.4)	**74.8 (0.4)**
	κ	43.5 (0.4)	61.0 (0.9)	58.4 (0.5)	55.8 (0.7)	**67.6 (0.5)**

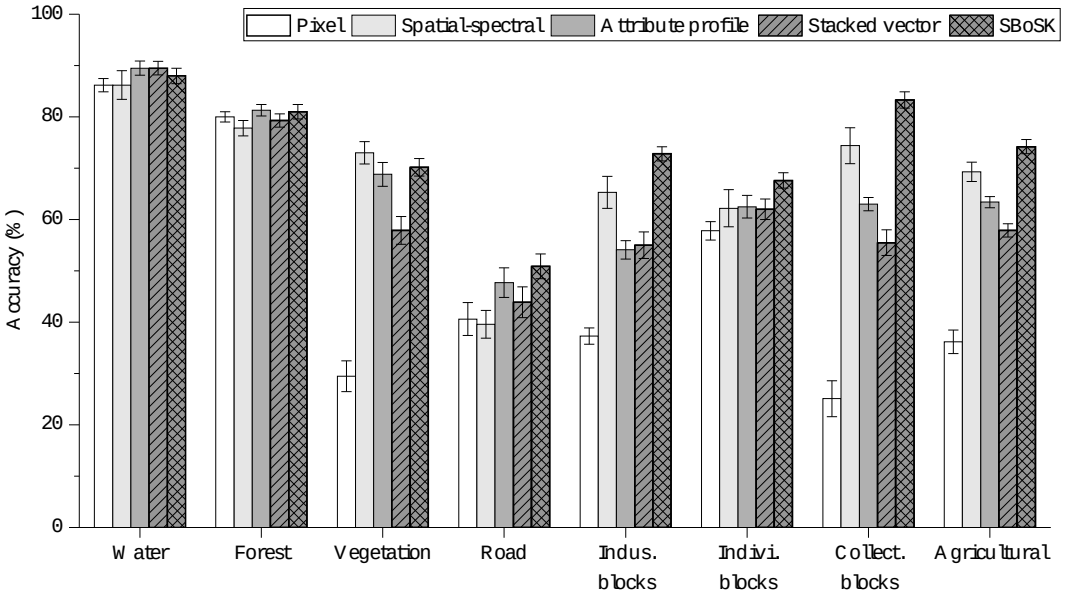


Figure 3.19: Per-class accuracies on Strasbourg MSR image.

3.6 Hyperspectral images classification

3.6.1 Datasets and design of experiments

We conduct experiments on 6 standard hyperspectral image datasets: Indian Pines, Salinas, Pavia Centre and University, Kennedy space center (KSC) and Botswana [1], considering a one-against-one SVM classifier [31].

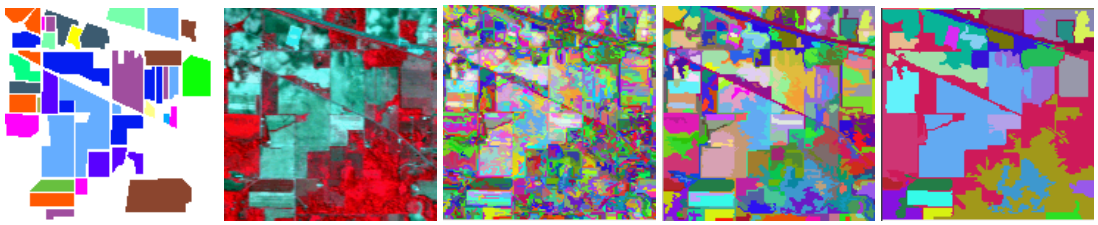


Figure 3.20: Indian Pines dataset. From left to right: ground truth, false-color image, fine level (2486 regions), intermediary level (278 regions), coarse level (31 regions).

We randomly pick $n = \{10, 25, 50\}$ samples per class for training, and the rest for testing. In the case of small number of pixels per class in Indian Pines dataset (total sample size for a class less than $2n$), we use half of the samples for training.

[1] The datasets descriptions and the associated ground truth are available at http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

We use the exact computation of BoSK in this section, as it is efficient when the training sample size is small. For its computation, we use Gaussian kernel as the atomic kernel $k(\cdot,\cdot)$. Free parameters are determined by 5-fold cross-validation over potential values: the bandwidth γ and the SVM regularization parameter C. We also cross-validate the different weighting scheme parameters: $P \in \{1, 2, \ldots, P\}$ for the maximum considered subpath length and $\lambda \in (0, 1)$ for the decaying factor in exponential weighting. All results are obtained by averaging the performances over 10 runs of (identical among the algorithms) randomly chosen training and test sets.

Hierarchical image representations are generated with Hseg by increasing the region dissimilarity criterion α. It is empirically chosen as $\alpha = [2^{-2}, 2^{-1}, ..., 2^{8}]$, leading to a tree that covers the whole scales from fine to coarse (top levels of whole image are discarded as they do not provide any additional information). Hierarchical levels $\alpha = \{2^2, 2^4, 2^6\}$ of Indian Pines are shown in Fig. 3.20 as the fine, intermediary, and coarse levels for illustration. Features $\boldsymbol{x}_{n_i}$ that describe each region are set as the average spectral information of the pixels that compose the region.

3.6.2 Results and analysis

We compare BoSK with state-of-the-art algorithms as done in the previous section: i) spatial-spectral kernel [61]; ii) attribute profiles [48], using 4 first principal components with automatic level selection for the area attribute and standard deviation attribute; iii) hierarchical features stored on a stacked vector [25, 113, 96]. For comparison purposes, we also report the overall accuracies of pixel-wise classification using only spectral information.

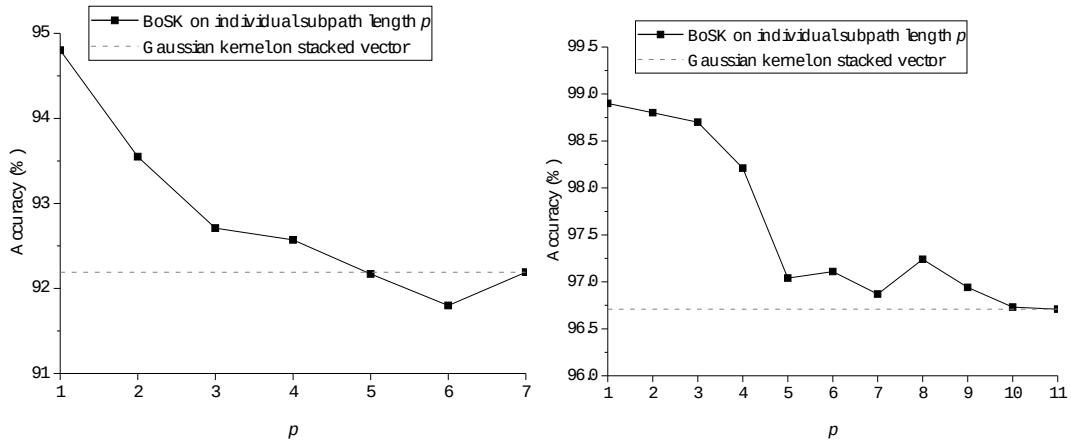
First of all, in Tab. 3.4, we can see that the overall accuracies are highly improved when contextual information is included. Using hierarchical features computed over a tree (*i.e.* stacked vector) yields competitive results compared with state-of-the-art methods. By applying BoSK on the same contextual information rather than the Gaussian kernel on stacked vector, the results are further improved: best results for Indian Pines, Salinas, Pavia Centre, KSC and Botswana datasets are obtained with BoSK. We observe that attribute profiles perform better for Pavia University. This might be due to the kind of hierarchical representation used, *i.e.* min and max-trees in the case of attribute profiles instead of Hseg in our case. Besides, the popularity of these profiles as well as the Pavia dataset result in optimizations of the scale parameters for years.

As far as the different weighting strategies are concerned, we see a further accuracy improvement using the exponential weighting or the maximum considered length weighting *w.r.t.* the constant weighting strategy. This observation on real world remote sensing datasets is consistent with the previous analyses on the synthetic dataset conducted in Sec. 3.4.2.

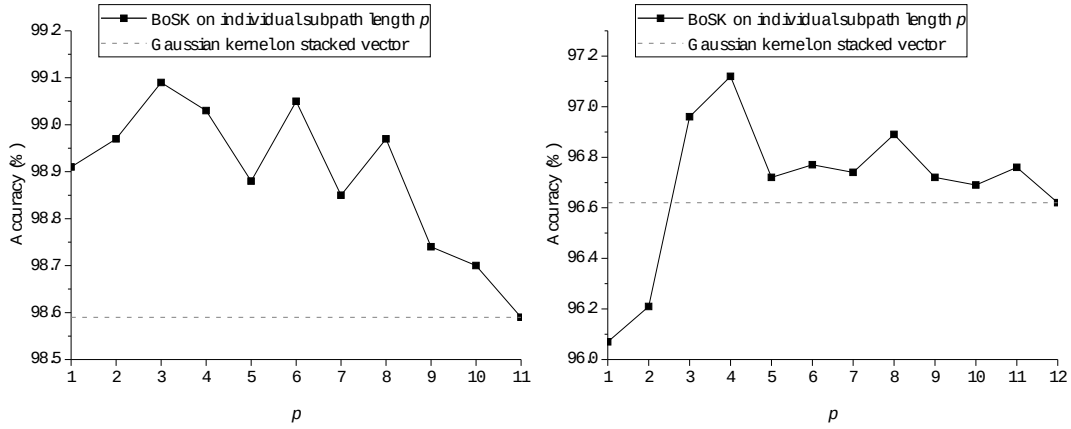
Fig. 3.21 shows the results of BoSK built on individual subpath length p. As one might notice, building the kernel on the longest subpath (corresponding to the Gaussian kernel on stacked vector) does not lead to the best performances, but yields one of the worst results in most cases.

Table 3.4: Mean (and standard deviation) of overall accuracies (OA) computed over 10 repetitions using n training samples per class for 6 hyperspectral image datasets. c stands for the constant weighting, λ for the exponential decaying weight, and P for the maximum considered length. Best results are boldfaced.

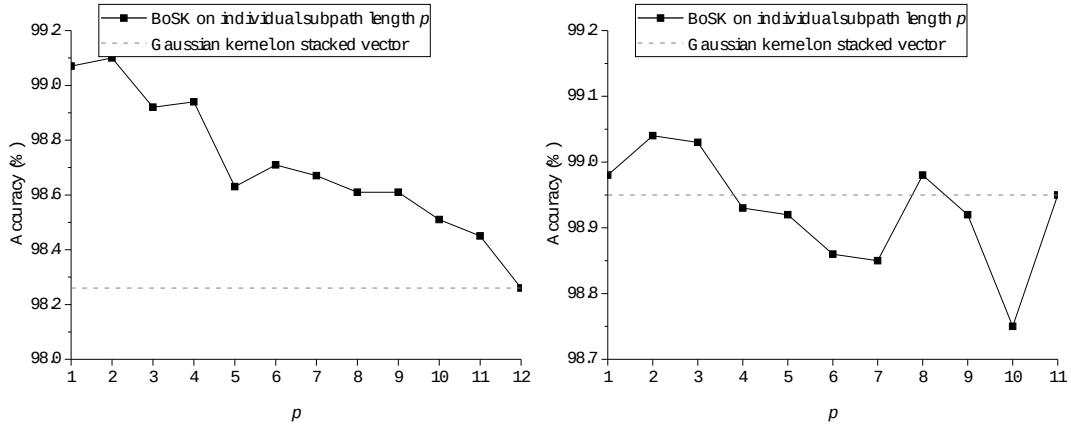
	Indian Pines						
n	pixel only	Spatial-spectral	Attribute profile	Stacked vector	BoSK-c	BoSK-λ	BoSK-P
10	54.89 (2.10)	72.03 (2.52)	64.37 (2.87)	73.21 (2.60)	78.70 (4.88)	80.19 (3.40)	**81.43 (2.39)**
25	66.04 (1.59)	84.02 (1.31)	76.71 (2.60)	84.90 (2.42)	89.16 (2.89)	89.46 (3.61)	**91.10 (1.84)**
50	72.99 (0.10)	90.82 (2.07)	84.57 (1.45)	92.19 (0.86)	94.12 (1.18)	94.48 (1.20)	**94.98 (1.01)**
	Salinas						
n	pixel only	Spatial-spectral	Attribute profile	Stacked vector	BoSK-c	BoSK-λ	BoSK-P
10	83.87 (1.96)	87.72 (1.88)	91.89 (1.73)	89.17 (2.95)	93.18 (1.70)	91.44 (2.71)	**93.84 (2.49)**
25	88.13 (1.22)	92.93 (0.98)	95.99 (1.11)	94.86 (1.58)	97.28 (1.62)	97.02 (1.57)	**98.22 (0.63)**
50	88.86 (1.22)	94.34 (0.81)	97.39 (0.45)	96.71 (0.70)	98.51 (0.89)	97.93 (1.22)	**99.00 (0.48)**
	Pavia Centre						
n	pixel only	Spatial-spectral	Attribute profile	Stacked vector	BoSK-c	BoSK-λ	BoSK-P
10	93.37 (3.59)	95.69 (0.73)	96.03 (0.91)	95.94 (1.01)	96.14 (1.61)	**96.71 (0.97)**	96.57 (1.02)
25	96.13 (0.48)	96.99 (0.48)	97.59 (0.27)	97.85 (0.53)	97.93 (0.55)	97.93 (0.57)	**97.95 (0.52)**
50	96.98 (0.52)	98.10 (0.34)	98.59 (0.24)	98.59 (0.48)	98.83 (0.39)	**99.04 (0.31)**	99.00 (0.30)
	Pavia University						
n	pixel only	Spatial-spectral	Attribute profile	Stacked vector	BoSK-c	BoSK-λ	BoSK-P
10	69.00 (5.68)	76.74 (5.26)	**88.69 (4.06)**	83.30 (3.75)	84.34 (5.14)	85.10 (6.65)	84.87 (5.01)
25	79.81 (1.42)	87.92 (3.36)	**95.17 (1.84)**	92.95 (3.29)	93.70 (2.56)	93.98 (2.22)	94.70 (2.17)
50	84.72 (1.32)	93.27 (1.29)	**97.52 (0.86)**	96.62 (1.06)	97.20 (0.97)	96.66 (1.84)	97.24 (1.03)
	KSC						
n	pixel only	Spatial-spectral	Attribute profile	Stacked vector	BoSK-c	BoSK-λ	BoSK-P
10	86.56 (1.33)	90.96(2.12)	90.61 (0.63)	92.75 (1.71)	93.98 (1.29)	94.01 (1.15)	**94.72 (1.18)**
25	91.27 (0.84)	97.16 (0.16)	95.53 (0.71)	97.32 (0.45)	97.85 (0.63)	97.82 (0.66)	**98.11 (0.64)**
50	93.67 (0.58)	98.46 (0.29)	97.41 (0.49)	98.26 (0.37)	99.13 (0.34)	99.15 (0.23)	**99.16 (0.32)**
	Botswana						
n	pixel only	Spatial-spectral	Attribute profile	Stacked vector	BoSK-c	BoSK-λ	BoSK-P
10	87.72 (2.42)	92.62 (1.40)	92.17 (1.32)	94.16 (1.41)	94.66 (1.62)	94.63 (1.54)	**94.94 (1.56)**
25	91.89 (0.67)	96.65 (0.69)	95.35 (0.91)	97.71 (0.72)	97.99 (0.48)	97.90 (0.79)	**98.00 (0.80)**
50	94.03 (0.60)	97.74 (0.52)	96.83 (0.64)	98.95 (0.53)	99.10 (0.50)	98.99 (0.45)	**99.28 (0.19)**



(a) Indian Pines (b) Salinas



(c) Pavia Centre (d) Pavia University



(e) KSC (f) Botswana

Figure 3.21: The overall accuracies of BoSK built on individual subpath length p using $n = 50$ training samples per class.

3.7 Large-scale image classification on Zurich summer dataset

In this section, we evaluate SBoSK on one large-scale publicly available dataset: "Zurich Summer v1.0" [204]. The dataset is a collection of 20 images, taken from a Quickbird acquisition of the city of Zurich with pansharpened resolution of about 0.62 cm. The images are composed of 4 channels (NIR, R, G, B), with an average image size of ca. 1000×1150 pixels. Examples of the dataset (images $16-20$ with associated ground truth in 8 different annotated urban classes) are shown in Fig. 3.22.

The term large-scale refers to a large number of training samples (each pixel in the image is considered as a data instance to be classified) for Zurich summer dataset with more than ten thousands data instances for training, and several millions for testing. These numbers are considered as large-scale in the context of classification using structured kernel, where evaluated datasets are normally made of thousands of data instances [128]. For this dataset, SBoSK is be applied with RFF dimension being empirically set at $D = 4096$.

For each image, we generate from the bottom level of each single pixel 6 additional levels of hierarchical segmentation with the Hseg segmentation tool using the region dissimilarity criterion $\alpha = [2^0, 2^1, \ldots, 2^5]$. Each region in the hierarchical representation is described by a 24-dimensional feature vector: the min, max, average and standard deviation values of the pixels included in the region for each spectral band and two derived channels: NDVI and NDWI (the Normalized Difference Water Index). As such, we use the same feature set as in [196].

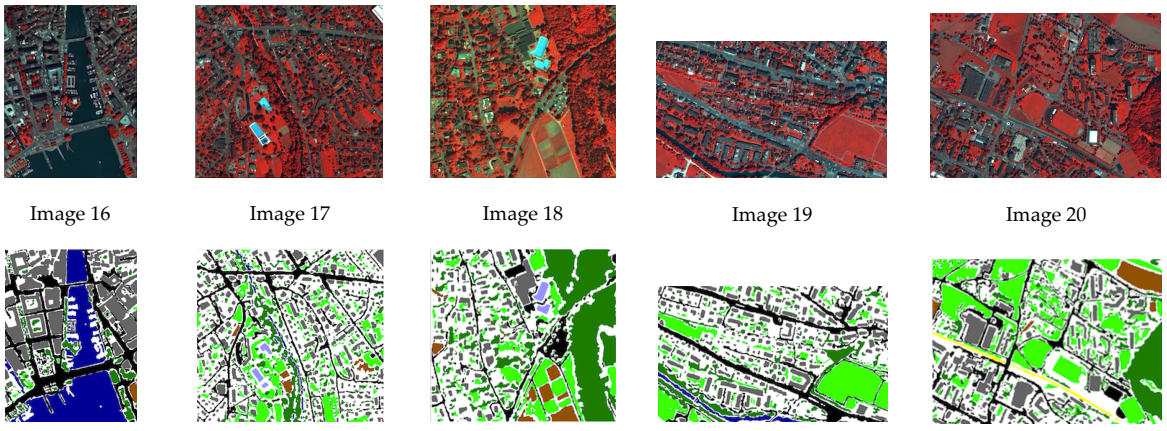
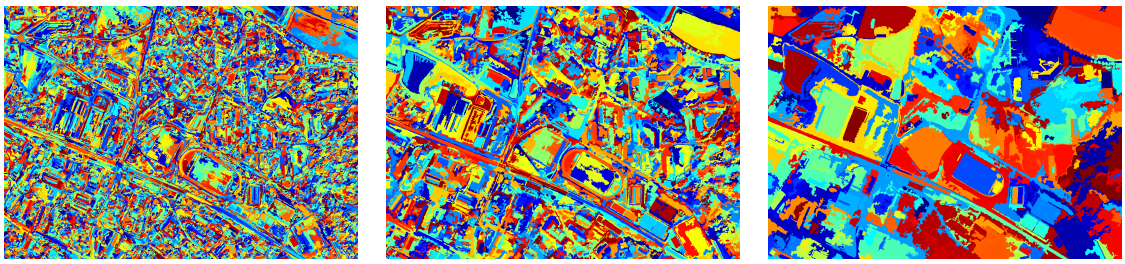


Figure 3.22: Examples of images $16-20$ (top row) in Zurich dataset. The associated ground truth (bottom row) with 8 different annotated urban classes: roads ■, buildings ■, trees ■, grass ■, bare soil ■, water ■, railways ■ and swimming pools ■.



(a) L2 (b) L4 (c) L6

Figure 3.23: Hierarchical segmentation maps of image 20 in Zurich summer dataset. 3 different levels of segmentation, L2,L4,L6, are illustrated here by using dissimilarity criterion $\alpha = [2^1, 2^3, 2^5]$.

Table 3.5: Mean (and standard deviation) of overall accuracies (OA), average accuracies (AA) and Kappa statistics (κ) computed over 10 repetitions for Zurich summer dataset images $16-20$. Best results (with a statistical significance less than 0.01% *w.r.t.* others considering the Wilcoxon signed-rank test for matched samples) are boldfaced, and numbers with * indicate that no statistically significant conclusions can be driven when compared with best results.

image		Pixel	CRF	Spatial-spectral	Attribute profile	Stacked vector	SBoSK
16	OA	71.8 (0.8)	82.8	81.6 (0.9)	78.5 (0.6)	83.4 (0.6)*	**83.9 (0.5)**
	AA	63.7 (2.1)	-	62.6 (1.1)	62.3 (0.8)	68.3 (1.1)	**70.8 (0.4)**
	κ	62.4 (0.8)	76.0	74.7 (1.2)	71.0 (0.6)	77.0 (0.9)*	**77.7 (0.7)**
17	OA	75.1 (0.7)	82.6	80.3 (0.6)	80.7 (0.9)	82.1 (0.6)	**83.2 (0.6)**
	AA	61.2 (3.6)	-	66.3 (1.8)	60.8 (1.9)	65.3 (1.6)	**67.7 (3.3)**
	κ	68.1 (1.0)	77.0	74.4 (0.8)	74.8 (1.2)	76.6 (0.8)	**78.1 (0.8)**
18	OA	81.1 (0.8)	73.0	85.1 (0.7)	83.1 (1.4)	85.7 (0.6)	**87.5 (0.3)**
	AA	74.0 (3.1)	-	78.6 (1.2)	74.5 (3.5)	78.6 (1.6)	**82.4 (0.6)**
	κ	72.4 (1.2)	62.0	77.8 (1.0)	74.7 (2.2)	78.5 (1.0)	**81.2 (0.5)**
19	OA	69.7 (0.7)	67.5	72.1 (1.8)	**78.4 (1.2)**	74.8 (0.6)	76.0 (0.6)
	AA	71.5 (0.9)	-	77.2 (1.5)	**80.4 (2.3)**	76.2 (2.9)	79.6 (1.4)*
	κ	61.1 (0.9)	57.0	64.0 (2.2)	**71.7 (1.6)**	67.1 (0.8)	68.8 (0.8)
20	OA	76.9 (1.1)	80.2	83.6 (0.9)	81.2 (1.2)	82.2 (1.2)	**84.0 (1.3)**
	AA	74.2 (1.2)	-	74.8 (1.4)	72.7 (2.1)	75.3 (4.8)	**77.4 (2.4)**
	κ	70.4 (1.3)	74.0	78.6 (1.2)	75.5 (1.6)	77.0 (1.5)	**79.3 (1.6)**
avg	OA	74.9 (0.6)	77.2	80.5 (0.5)	80.4 (0.7)	81.7 (0.4)	**82.9 (0.3)**
	AA	68.9 (1.8)	-	71.8 (0.6)	70.1 (1.5)	72.7 (1.2)	**75.6 (0.8)**
	κ	66. (0.8)	69.2	73.9 (0.7)	73.5 (0.9)	75.2 (0.5)	**77.0 (0.3)**

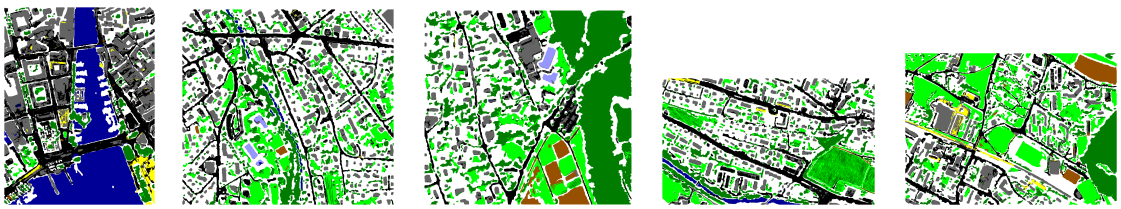


Figure 3.24: Classification maps of Images 16 to 20 of the Zurich summer dataset using SBoSK.

To allow a fair comparison with the state-of-the-art, we follow the experimental setup provided in [196]: we use the images $1-15$ solely for training (with a stratified selection of 0.1% available training samples per class, which corresponds to 12 , 263 pixels chosen from all training images), and the images $16-20$ only for evaluation. The final classification results are computed over 10 repetitions for random dataset splits into training and evaluation sets.

The overall accuracies, average accuracies and kappa index κ are shown in Tab. 3.5 for individual images $16-20$. We can see that the kernel computed at the single pixel using only spectral information yields the worst results *w.r.t.* methods taking into account contextual information. Comparing to the state-of-the-art method using Conditional Random Fields [196], building kernels on a hierarchical representation (*i.e.* spatial-spectral, attribute profiles, stacked vector) can provide a better result. As expected, SBoSK further improves the results achieved with stacked vector relying on the same paths, leading to the overall best results. Classification maps obtained with SBoSK are given in Fig. 3.24. As we can see, SBoSK produces spatially smooth classification maps, with most of the compact regions being correctly predicted.

3.8 Chapter summary

In this chapter, we presented the first application of (S)BoSK on the path structure for pixel-wise image classification. The path structure allows taking into account the spatial context of a pixel (leaf of the hierarchical representation) through its ancestral regions at multiple scales. (S)BoSK can take path structures as inputs and then exploit the regions at different scales and the hierarchical relationships among them. We also show that the Gaussian kernel on stacked vector is actually a special case of (S)BoSK *i.e.* considering only the maximal length of subpath.

The analysis has been done firstly with a synthetic dataset in order to illustrate the superiority of BoSK when compared to the Gaussian kernel on stacked vector, and also the performance of its scalable version SBoSK. Experimental results clearly show that BoSK performs better than the Gaussian kernel on stacked vector in all settings. In fact, the Gaussian kernel on stacked vector yields one of the worst performances among BoSK using other subpath lengths. As far as the scalability is concerned, we observed that the classification accuracy of SBoSK approximates the exact computation when using a reasonable Random Fourier Features dimension D, while the computation time reduces from quadratic to linear *w.r.t.* training data size.

We also used a real-world urban remote sensing classification task to further illustrate the superiority of BoSK and the advantages of using its scalable version SBoSK. Evaluations

show that SBoSK outperforms several state-of-the-art pixel-wise classification methods considering contextual information.

Such superiority of (S)BoSK is further confirmed with experiments on 6 standard publicly available hyperspectral image datasets. With Gaussian kernel on stacked vector, we achieved similar results *w.r.t.* state-of-the-art methods using context features extracted from hierarchical representation (*i.e.* spatial-spectral kernel and attribute profiles). SBoSK further improves the results achieved with stacked vector, yielding the best results in 5 out of 6 publicly available hyperspectral image datasets.

Finally, we evaluated the SBoSK in a large-scale classification context using a recent publicly available dataset. The classification accuracy obtained by SBoSK outperformed state-of-the-art methods using context features, as well as one recent proposed method that relies on Conditional Random Fields.

Chapter 4

Spatial decomposition-based sub-image/tile classification

Contents

In the previous chapter, we presented (S)BoSK for path structured data that encodes contextual information through ancestral regions, allowing one to perform the pixel-based classification.

We introduce in this chapter the second application of (S)BoSK for tree structured data. It classifies the root node that represents sub-image/tile image, and takes input of the tree structure that reveals the object spatial decomposition.

4.1 Introduction

Sub-image/tile images classification has been intensively studied over the last decades in the domain of computer vision [109, 147, 97] and remote sensing [213, 143, 220, 221, 216]. Due to the large covering of image content, especially in the context of remote sensing, current approaches concentrate on splitting the observed scene taken from large Earth surface into small tiles, and process the image of each tile independently. These methods focus on developing the sub-image/tile image classification strategies, and refer them using the general term of image classification instead [220, 216]. In this chapter, we follow this convention and use the term image classification.

The main objective of image classification is to assign each image to one of a list of predefined classes according to its content. Global image features such as image texture [89, 172, 86] or color histogram [84, 153] are popular thanks to their simplicity. However, they do not always achieve good results. This is due to their coarse description of image, which might be sensible to viewpoint and lighting changes, scales, clutter and occlusions [217, 34]. Instead, local features such as SIFT [123] have received increasing attention during the last decade and are probably the main trend for image classification. The basic pipeline of image classification using local features is generally called "Bag of visual Words (BoW)" [33], which includes: local features extraction, local features quantization, and histogram representation of quantized local features (also known as feature encoding). Although various studies have been proposed for improving the classification accuracy by using advanced feature encoding methods *e.g.* Fisher vector [154] or VLAD [54], one fundamental limitation of the BoW is the orderless organization of extracted local features, where the spatial relationships among the local features are no longer preserved. Another limitation is related to its description capability, as these extracted local features are often considered as low-level features *e.g.* image gradient orientation over a small size window.

Instead of using low-level interest point-based features, region-based image classification have been proposed [112, 8, 200, 111, 87]. In this framework, regions are constructed by grouping together the similar pixels according to some homogeneity measure. In order to generate perceptually meaningful entities inside an image, advanced image segmentation techniques are often used [132, 80]. After construction, regions are described by a rich set of features, *i.e.* shape, color, texture and even group of local features such as SIFT [221]. Moreover, the spatial relationships among regions can be also preserved [77], reflecting the inner structure of the images. However, the segmentation into meaningful and precise regions is very difficult, as the regions definition is highly subjective to the classification problem at hand. For instance, the "ideal" segmentation for a urban area classification task (*e.g.* residential area or industrial area) must be very different than the classification of road network, vehicle or building, the later requiring smaller regions that can cover potential objects.

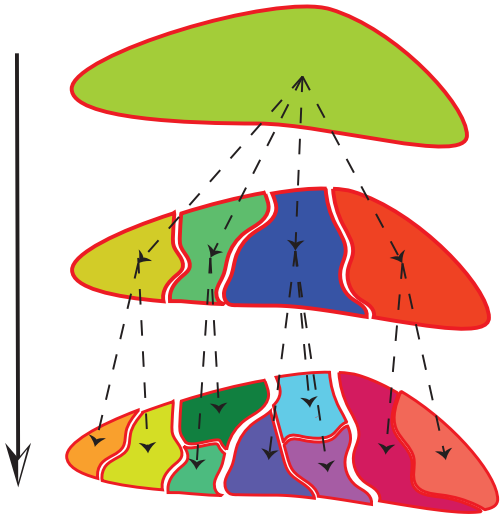


Figure 4.1: Illustration of a hierarchy of objects generated from a hierarchical image representation, and the spatial decomposition of root node through its subregions and the topological relationships among them.

Therefore, there is no optimal solution for segmentation that can fit for various classification tasks. In addition, objects-of-interest might appear at different scales (level of detail). This is even harder to define the scale parameter for segmentation.

An emerging paradigm for image classification advocates the idea of relying on hierarchical representations [17], which are built using series of nested partitions or segmentations, rather than on the usual flat representation. Regions at different scales are generated using multiscale segmentation tools and represented through a tree structure, where the root node represents the whole image and the leaves stand for the finest scale of segmentation. Regions are the nodes of the tree described by a set of features as the nodes attributes, and the relationships among regions are modeled through the edges, as illustrated in Fig. 4.1.

In a hierarchical representation, spatial decomposition information can be revealed. It models the composition of an object and the topological relationships among its subparts. Including such information can improve the classification rate, for instance, a residential area is much easier to be identified when knowing it is composed of houses and roads. It is especially true in high resolution remote sensing imagery cases where decomposition of objects can be better revealed [35, 221].

In this chapter, we propose an image classification approach based on object spatial decomposition. Such information is modeled through hierarchical representations and is represented as a tree structured data, on which (S)BoSK can be directly applied. The chapter is organized as follows: a brief review of related work is provided in Sec. 4.2. In Sec. 4.3,

we describe the proposed classification method using (S)BoSK, which is followed by detailed analysis on a synthetic dataset in Sec. 4.4, and evaluations on remote sensing dataset in Sec. 4.5 and Sec. 4.6.

4.2 Related work

Image classification approaches using topological information of image regions are the main focus in this chapter. In order to position our proposed method in literature, we first give a brief review of the methods using structured kernel for capturing the topological information, then focus on the approaches learning on hierarchical representations.

4.2.1 Capturing topological information using structured kernel

Several methods that represent image regions and relationships among them as a structured data and rely on structured kernel to perform classification have been proposed in literature. In [124], the images are firstly segmented into a regular grid with each block being quantized into one of the visual keywords, then the images are represented as a 2D sequence of symbolic blocks. Mismatch string kernels dedicated to symbolic data [115, 207] are applied in order to capture the spatial dependencies of the generated blocks across an image.

Graph kernel is also used for considering the layout of regions within an image. Such topological information is revealed through region adjacency graph (RAG), where the regions are generated using an image segmentation method and are represented as the nodes, and relationships among regions are modeled in the edges. Path [8, 200, 111, 110] and subtree patterns [87] inside RAG are extracted and learned with graph kernel for considering the high-level topological information. Similarly, these graph kernels have also been applied in skeleton graph for shape recognition [57, 177, 179].

The aforementioned graph kernels are closely related to our context. However, their application to tree structures is not straightforward. From a structured kernel construction point of view, the graph kernel might use a complex substructures such as subtree pattern [87], while it might yield computational issues with unordered trees. Other kernels such as marginal graph kernel [177, 179] rely strongly on the properties of the graph, and their selection of substructures and computation scheme can not be directly applied.

From an application point of view, the major differences come with the fact that image graphs are most often rootless, undirected, and bring a flat representation of the image (*i.e.*, with a single spatial scale). The classification accuracy based on planar region might be limited by the quality of the segmentation, as meaningful and highly precise regions are required.

4.2.2 Topological information in hierarchical image representations

An emerging paradigm for image classification advocates the idea of relying on hierarchical representations [17] and has gained increasing attention in the remote sensing community. However, the hierarchical relationships among objects are often captured through manual semantic modeling, which requires prior knowledge based on human interpretation to derive proper classification rules [58, 9].

Among studies that use machine learning techniques to learn on regions and their hierarchical relationships, [82] relies on regions that are generated from multiscale segmentation, and applies discriminative max-margin framework to learn the region weights representing the importance of each region *w.r.t.* classes. Hierarchical spatial structure patterns extracted from cross-scale regions are modeled in [7, 211, 156, 215] within a probabilistic graphical model framework, where the pairwise relationships between regions is modeled through the definition of the potentials, allowing one to model hierarchical relationships among regions. The previous methods tend to learn the importance of the regions or spatial structures through various learning algorithms. However, in order to guarantee a good classification performance, correct weights need to be learned and this might require a large amount of training data. In addition, the complexity of modeling hierarchical structures limits the methods to small substructures.

The Spatial Pyramid Matching (SPM) model [109, 209] is the most common strategy to consider the object spatial decomposition. It often relies on a kernel based machine learning algorithm to perform image classification tasks. The idea is to segment the image in 4 regions at successive scales (quad-tree representation as illustrated in Fig. 4.3a), and to concatenate all the region features into a long vector. However, SPM only allows matching image regions at the same spatial position. Therefore, applying SPM in remote sensing image classification tasks raises some severe issues since SPM hardly adapts to images with no predefined location or orientation [221, 35, 213]. Recently, methods such as the pyramid of spatial relatons [35] have been proposed for tackling these issues. However, the matching strategy of SPM limits its application to quad-tree representations only, thus preventing the benefits of available advanced multiscale segmentation techniques able to produce a wide range of tree topologies.

In this chapter, we use (S)BoSK on an unordered tree structure for performing spatial decomposition-based image classification. It can be applied for arbitrary hierarchical representations, and it is robust to image rotation and translation. We describe our approach in more details in the following sections.

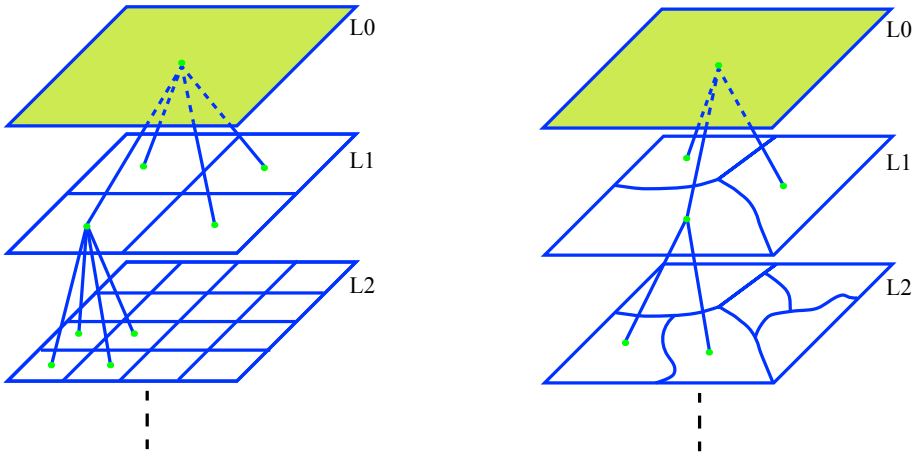
4.3 (S)BoSK on object spatial decomposition

We rely on (S)BoSK to classify images. Each image is represented hierarchically, where the top level stands for the whole image and finer levels reveal the detailed information of the image. An example of such a hierarchical organization of objects-of-interest is shown in Fig. 4.2, where larger regions are iteratively divided into smaller ones across the levels. The constructed hierarchical representation can be expressed as a tree structure $\mathcal{T}$, where objects-of-interest are the nodes in the tree and the hierarchical relationships among them are modeled through the edges (Fig. 4.3). (S)BoSK can be thus applied on the induced tree structures.

The hierarchical image representations can be constructed either by iteratively segmenting the image in 4 regions at successive scales (quad-tree representation as in Fig. 4.3a), or by multiscale segmentation algorithms (as shown in Fig. 4.2 and Fig. 4.3b).

(a) Image (b) level 0 (c) level 1 (d) level 2 (e) level 3 (f) level 4

Figure 4.2: Illustration of multiscale image segmentation from level 0 (whole image) to level 4.



(a) Quad-tree representation used in the Spatial Pyramid Matching Model

(b) Hierarchical representation generated with a multiscale segmentation algorithm

Figure 4.3: Illustration of a quad-tree representation and an arbitrary hierarchical representation.

From the constructed hierarchical representation, object spatial decomposition can be easily revealed. It is modeled by the composition of objects and the topological relationships

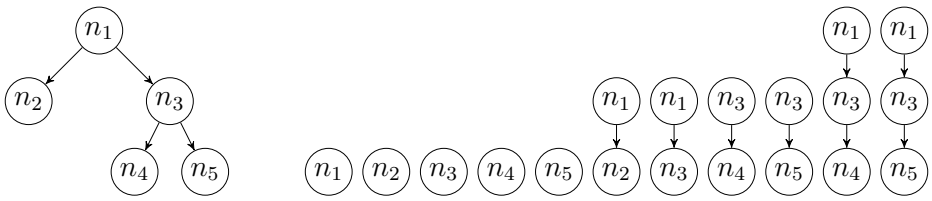


Figure 4.4: An example of a tree $\mathcal{T}$ and its subpaths s_p.

among their subparts. For instance, an image of a harbor area is composed of water and group of boats at coarse levels, and the group of boats is divided into individual boats at intermediary levels, which is further separated in different parts of boat at finer scales. An example of such an object decomposition through a hierarchical representation is shown in Fig. 4.2.

The object spatial decomposition can be represented as a tree structure, and can be further taken into account by (S)BoSK. Indeed, (S)BoSK operates on subpaths, allowing one to capture the nodes and vertical hierarchical relationships between the nodes. An example of a tree and its set of subpaths is shown in Fig. 4.4, where we can see that the regions, pairwise parent-child region pairs, and even longer patterns of the hierarchical relationships among regions are all included in the subpath set. By matching all the subpath structures representing the object decomposition patterns, (S)BoSK computes the similarity between two hierarchical representations. More specifically, (S)BoSK between two tree structures $\mathcal{T}, \mathcal{T}'$ is written as:

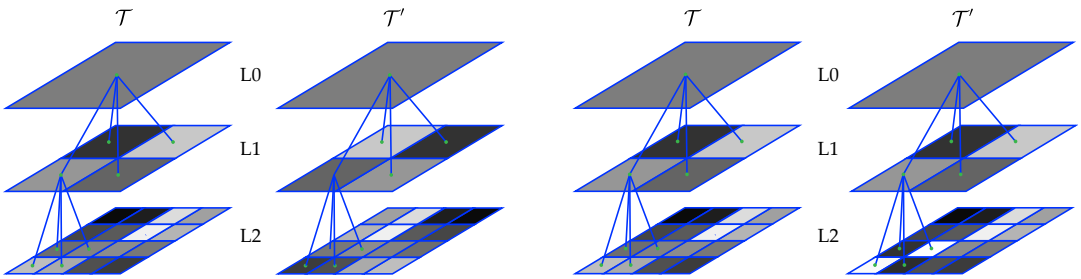
$$K(\mathcal{T}, \mathcal{T}') = \sum_{p=1}^{P} \mu_p \sum_{s_p \in \mathcal{T}} \sum_{s'_p \in \mathcal{T}'} K(s_p, s'_p) \,, \tag{4.1}$$

where $K(s_p, s'_p)$ is computed over all pairs of subpaths of same length extracted from two hierarchical representations and μ_p associates a weight for different subpath lengths.

Applying (S)BoSK on hierarchical image representations offers several advantages, since (S)BoSK is

- invariant to image rotation, since the subpaths pairs can be extracted from different spatial locations of images;
- robust to image scale change, since the matching of subpath pairs from different scales of hierarchy is also allowed;
- robust to image partial changes, since the sum operations of (S)BoSK can yield a similarity that is proportional to the number of similar subpath pairs.

We now illustrate the advantages of (S)BoSK using two straightforward examples. We construct two quad-tree representations $\mathcal{T}$ and $\mathcal{T}'$ with region feature being the gray level (in



(a) Two similar quad-tree representations with rotation.

(b) Two similar quad-tree representations with partial changes.

Figure 4.5: Example of similar quad-tree representations with a certain level transformation.

Baseball diamond Harbor

(a) Two similar images with rotation.

Run way Storage tanks

(b) Two similar images with partial changes.

Figure 4.6: Example of intra-class similar images with a certain level transformation.

Fig. 4.5). To compare BoSK and Spatial Pyramid Matching (SPM) kernel, we use a Gaussian kernel with high gamma value *i.e.* $\gamma = 10000$ that equals to 1 when two region gray levels are similar, otherwise close to 0. SPM is computed using a Gaussian kernel between two vectors $K_{SPM}(\mathcal{T}, \mathcal{T}') = k(H_{\mathcal{T}}, H_{\mathcal{T}'})$, where $H_{\mathcal{T}}$ (*resp.* $H_{\mathcal{T}'}$) is the concatenation of all region features for tree $\mathcal{T}$ (*resp.* $\mathcal{T}'$) . Note that we have self-similarity equals to 1 for both kernels: $K_{SPM}(\mathcal{T}, \mathcal{T}) = 1$, $K_{BoSK}(\mathcal{T}, \mathcal{T}) = 1$, (*resp.* $\mathcal{T}'$).

In the first example (in Fig. 4.5a), $\mathcal{T}$ and $\mathcal{T}'$ are almost identical but the order of the children is different. This corresponds to image rotation (an example in the remote sensing context is shown in Fig. 4.6a). After applying SPM and BoSK, we have $K_{BoSK}(\mathcal{T}, \mathcal{T}') = 1$, as $\mathcal{T}$ and $\mathcal{T}'$ are identical orderless trees. However, for SPM, we have $K_{SPM}(\mathcal{T}, \mathcal{T}') = 0$, indicating that the two trees are completely different.

In the second example (Fig. 4.5b), $\mathcal{T}$ and $\mathcal{T}'$ are similar as they share some certain parts

of similar content (an example in the remote sensing context is shown in Fig. 4.6). After applying the SPM and BoSK, we have $K_{BoSK}(\mathcal{T},\mathcal{T}') = 0.79$, as $\mathcal{T}$ and $\mathcal{T}'$ share a majority of their parts. However, for SPM, we have $K_{SPM}(\mathcal{T},\mathcal{T}') = 0$. Again it indicates that the two trees are completely different, which is obviously against the intuitive understanding of similarity.

4.4 Experiments on a synthetic dataset

We study here the behavior of the proposed (S)BoSK as a tree structured kernel though the following scenario using an artificial dataset. In Fig. 4.7, two classes consist of similar leaves at the bottom level. However, the spatial arrangement of these leaves generates new different nodes at intermediary level, also affecting vertical relationships between these nodes.

Such a scenario can be found in the remote sensing image classification context, especially when the classes correspond to complex (non-uniform) patterns. For instance, given two classes as individual residential area and collective residential area, they are both composites of tree species and buildings. However, in the individual residential area class, tree species are more likely to be merged with building at intermediary level, which will form the new regions of mixed tree species and buildings, while for collective residential area, tree species and building are more likely to be merged together separately at intermediary level, then the group of tree species and buildings are joined in the end. In this configuration, the two classes share the similar objects at bottom level and root at the top level, but the intermediary level objects and vertical relationships among the subparts reveal the discriminative features between the two classes.

4.4.1 Dataset description and experimental setup

To simulate the concept described so far and generate the aforementioned two classes using similar strategy as in Sec. 3.4.1. We generate two classes as shown in Fig. 4.7 by forcing type A leaves to merge with type B leaves in `Class 1`, while in `Class 2`, type A (*resp.* B) leaves always merge with type A (*resp.* B). In our evaluation, we have for each tree about $80 \sim 120$ leaves and the depth is about $4 \sim 7$.

We consider a one-against-one SVM classifier with the Gaussian kernel as the atomic kernel. Three BoSK weighting strategies are considered: i) BoSK with constant weights; ii) the limit on the maximum length of substructures with $P \in \{1,2,\dots 7\}$; iii) the use of exponential weighting.

Accuracies (and standard deviations) of each setup are computed after 10 repetitions of each experiment, choosing randomly 20 data samples from each class as training samples, using 80 data samples for testing.

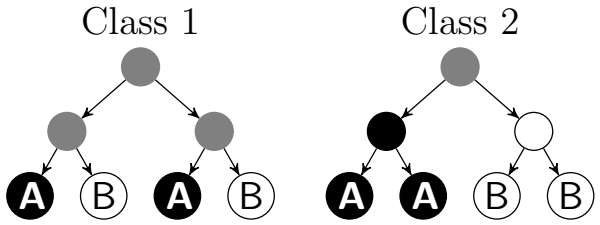


Figure 4.7: Synthetic concept for experimental evaluations.

When using directly the Gaussian kernel on the root nodes, we obtain an accuracy of about 50%, because of the non discriminative root of the two classes. However, the tree structure information, such as the discriminative nodes at intermediary levels and the hierarchical relationships between nodes through different levels of the tree, can be easily captured by BoSK (with all three weighting schemes), leading to a 100% accuracy.

4.4.2 BoSK analysis

Overall evaluation of BoSK

We study the behavior of BoSK by adding some confusion or noise inside the two classes. The same two particular behaviors have been studied as in Sec. 3.4.2: robustness to outliers and robustness to mislabeled leaves.

Fig. 4.8 and 4.9 present the accuracies obtained in these two settings. We can notice that in both scenarios, BoSK maintains a good performance up to a certain ratio of structure distortion (with all three weighting strategies). In addition, we observe that using different weighting strategies, such as exponential weighting and the use of maximum length for subpaths, can affect the results and that the later yields the best results in both scenarios.

Impact of the maximum considered subpath length

The impact of the maximum considered subpath length is analyzed here by using different $P \in \{1, 2, \dots 7\}$. We consider BoSK in both scenarios with a certain ratio of structure distortion: 60 %, 65 %, 70 % being chosen in case of outliers, and 25 %, 30 %, 35 % being chosen in case of mislabeled leaves, for ease of analysis.

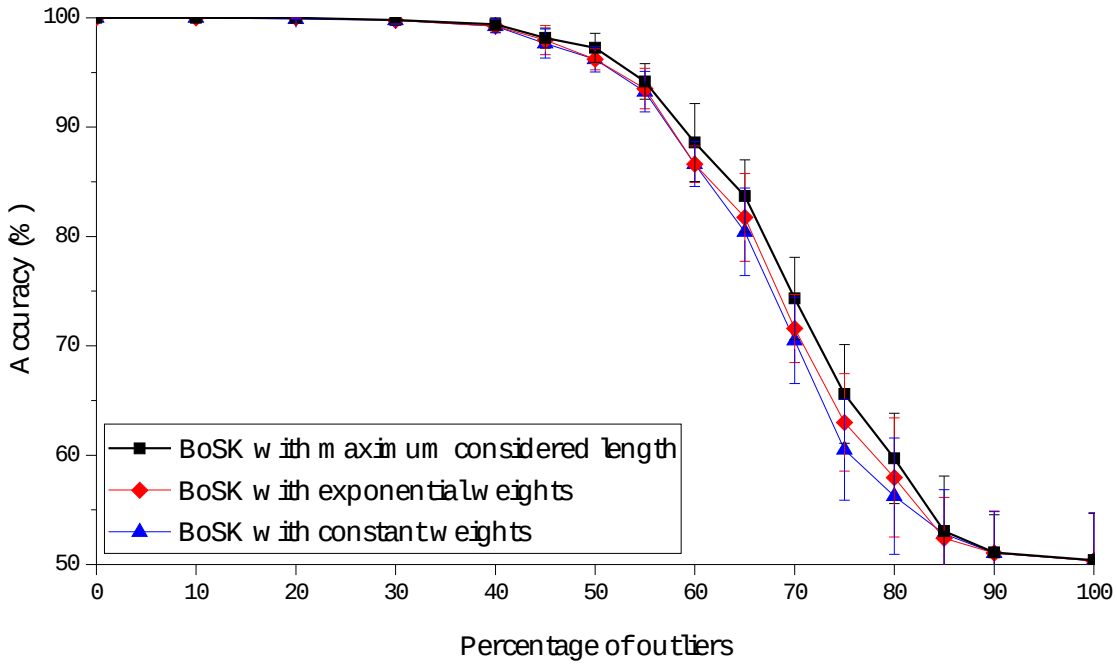


Figure 4.8: Accuracies and standard deviations of BoSK using different weighting strategies in presence of outliers.

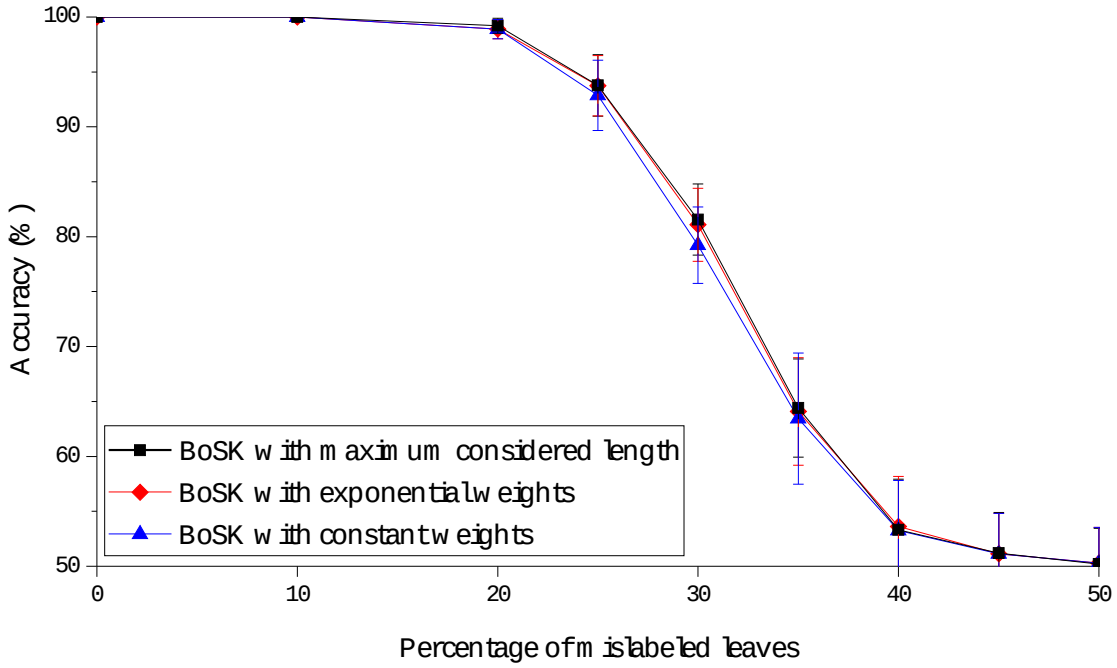


Figure 4.9: Accuracies and standard deviations of BoSK using different weighting strategies in presence of mislabeled leaves.

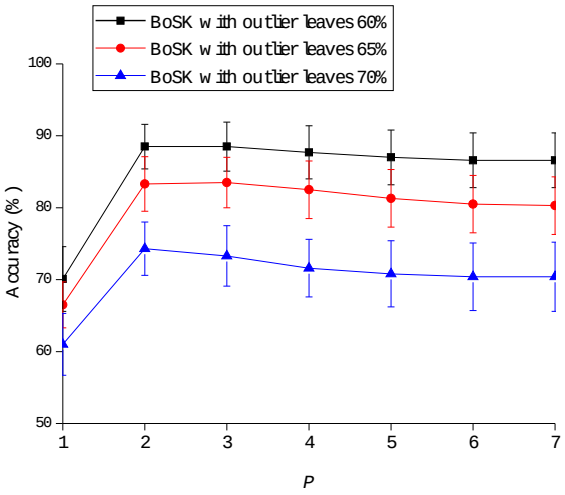


Figure 4.10: Impact of maximum considered subpath length in the case of outliers.

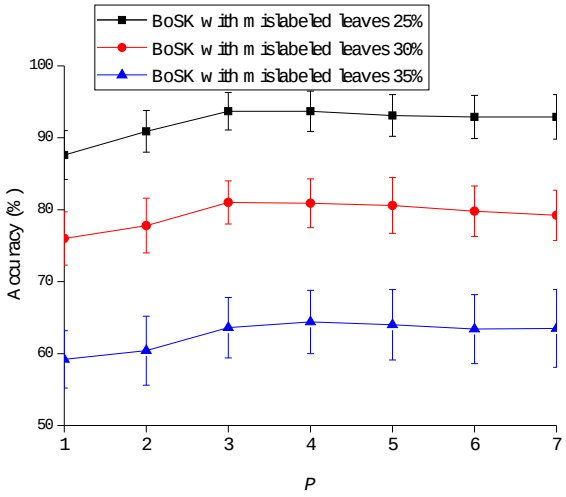


Figure 4.11: Impact of maximum considered subpath length in the case of mislabeled leaves.

Fig. 4.10 and Fig. 4.11 show that in case of outliers, the maximal accuracies are obtained for a combination of subpaths of length equal to 2, then the performances gradually decrease when adding longer subpaths to the global kernel value computation; while in case of mislabeled leaves, the maximal accuracies are obtained with longer subpaths (*e.g.* $P = 3$, $P = 4$), and performances drop down slightly with increasing length of subpaths. The observation of maximum accuracies being obtained within a limited number of lengths confirms again for penalization of longer subpath patterns.

4.4.3 SBoSK analysis

In this section, we study the behavior of the scalable version of BoSK. We consider the scenario of 30% mislabeled leaves, since this setting leads to a reasonable ratio of structure distortion between the two classes for the sake of analysis. The SBoSK analysis follows the same organization as in Sec. 3.4.3

Approximation error

Fig. 4.12 shows the relation between the kernel approximation error and the Random Fourier Features dimension D: when the dimension increases, the approximation will tend to zero with exponential convergence.

Classification accuracy

Fig.4.13 shows the classification accuracies of BoSK and SBoSK When the Random Fourier Features dimension D increases, the accuracy also increases until it converges to the accuracy

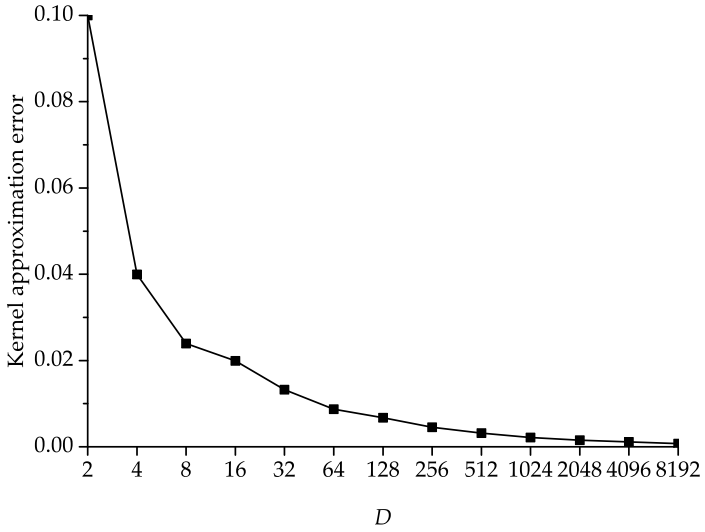


Figure 4.12: SBoSK approximation error *w.r.t.* Random Fourier Features dimension D (log scale).

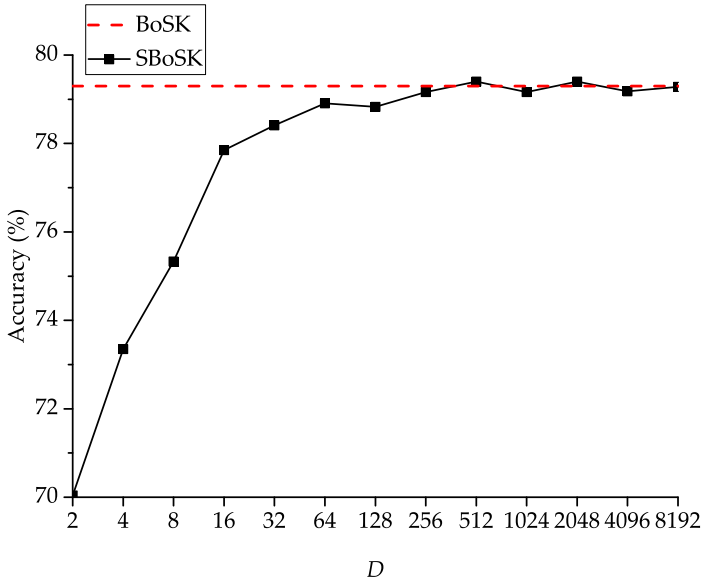


Figure 4.13: Classification accuracy *w.r.t.* Random Fourier Features dimension D (log scale).

obtained by BoSK. This agrees with the kernel error analysis shown in Fig. 4.12.

In addition, we analyze the L_2 normalization strategy on SBoSK. Fig. 4.14 shows that the accuracies improve greatly when considering subpath with different lengths *w.r.t.* SBoSK using only nodes ($P = 1$). However, the accuracies might decrease when adding the features extracted from longer subpath patterns, thus calling for penalization of longer subpath patterns. Note that the L_2 normalization used in SBoSK may result in a different accuracy plot than the one obtained by BoSK. However, they both achieved similar optimal accuracies using a maximum considered subpath length set to $P = 3$.

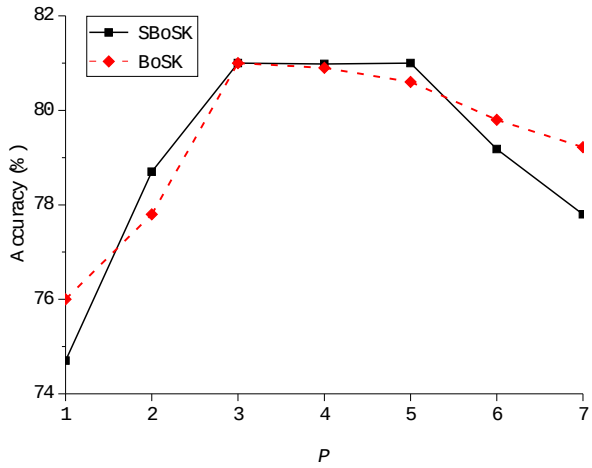


Figure 4.14: Classification accuracy of SBoSK *w.r.t.* maximum considered subpath lengths P.

Complexity analysis

Fig. 4.15 shows that the computational time increase linearly *w.r.t.* dimension of the Random Fourier Features $O(D)$ and Fig. 4.16a shows that the time for BoSK increases quadratically *w.r.t.* training sample size $O(n^2)$. In addition, we show here the advantage of SBoSK when dealing with a large tree size. Fig. 4.16b shows that the computational time increases quadratically $O(|\mathcal{T}|^2)$ for BoSK, while for SBoSK, it increases linearly $O(|\mathcal{T}|)$. This linear complexity makes SBoSK relevant for real-world applications with a large number of regions in the hierarchical representation.

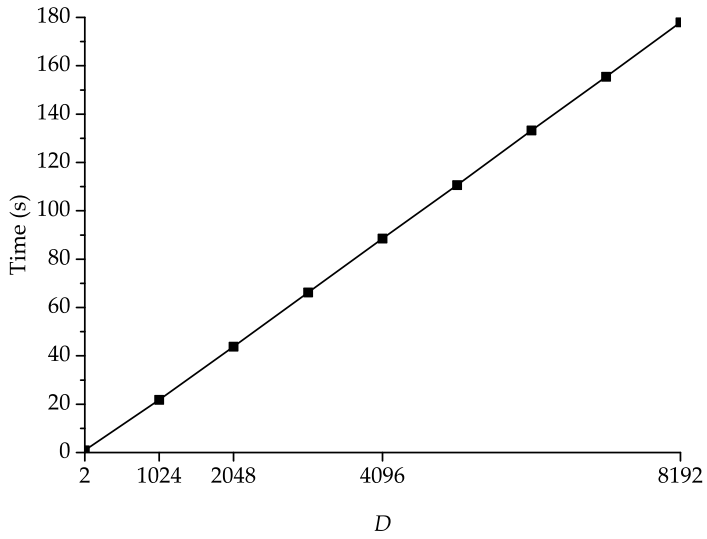
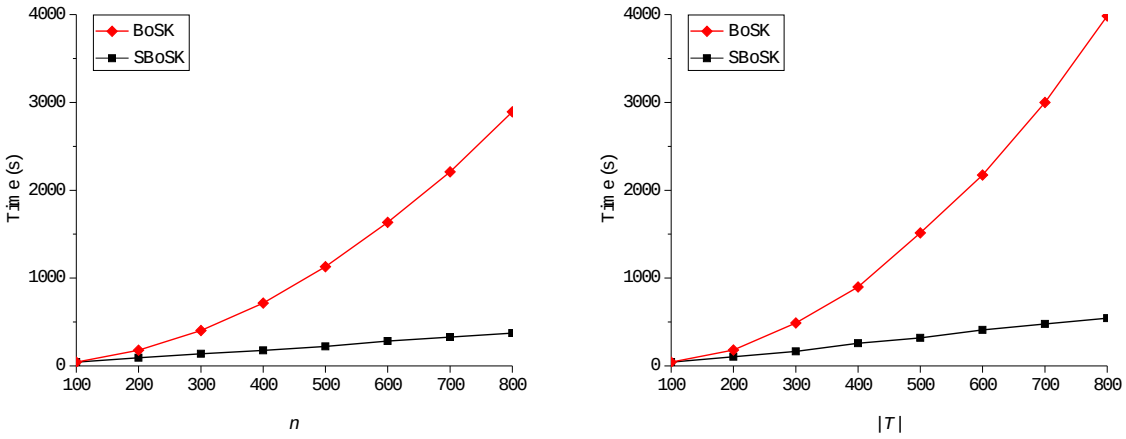


Figure 4.15: Computational time of SBoSK *w.r.t.* Random Fourier Features dimension D.



(a) (b)

Figure 4.16: Computational time of BoSK and SBoSK *w.r.t.* training samples size n (Fig. 4.16a) and *w.r.t.* tree size $|\mathcal{T}|$ (Fig. 4.16b).

4.5 Strasbourg Pleiades image classification

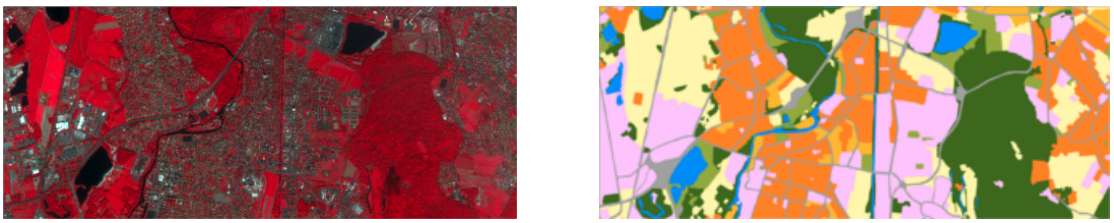
4.5.1 Datasets and design of experiments

In this section, we focus on urban land-use classification using VHSR image of 13040×5400 pixels with 0.5 m spatial resolution. The image has been acquired by the Pleiades satellite and is made of 4 spectral bands *i.e.* Red, Green, Blue, NIR. The image and it associated ground truth are shown in Fig.4.17. Note that we use the same ground truth as the evaluation in Strasbourg Spot-4 dataset, the 8 thematic classes of urban patterns are thus the same. Readers can find the full details of image description in [107].

On the Strasbourg Pleiades dataset, we generate the data instance to be classified as square regions of size 40×40 pixels. Two types of hierarchical representations are used: i) a pyramid representation with L2 level as the finest level, which corresponds to subregions of size 10×10 pixels. ii) a hierarchical segmentation generated using Hseg, where 4 additional levels are constructed by decreasing the region dissimilarity criterion $\alpha = [2^4, 2^3, 2^2, 2^1]$. Using such parameters, we observe an average of 16 leaves (the number of segmented regions at bottom level is then similar to Pyramid representation with the $L2$ level) and 30 nodes.

Each region in the hierarchical representation is described exactly as with Strasbourg Spot-4 dataset in Sec. 3.5, which includes the region average of the 4 original multi-spectral bands, Soil Brightness index (BI) and NDVI, as well as two Haralick texture measurements (homogeneity and standard deviation). Similarly, we use the Gaussian kernel as the atomic kernel to compute the similarity within a pair of nodes.

We consider a one-against-one SVM classifier with Gaussian kernel as the atomic kernel.



(a) Pleiades image (b) Ground truth image

Figure 4.17: Urban scene taken over South of Strasbourg, France. From left to right: false color image of Pleiades (© CNES 2012, distribution Airbus DS / Spot Image) with 50 cm resolution and the associated ground truth (© LIVE UMR 7362, adapted from OCSOL CIGAL 2012) with eight thematic classes.

All free parameters are determined by 5-fold cross-validation. The RFF dimension D is chosen empirically as a trade-off between computational complexity and classification accuracy (and will be further analyzed in Sec. 4.5.2). Henceforth, in this section, all reported results are averaged over 10 repetitions.

4.5.2 SBoSK analysis

The analysis of SBoSK in terms of classification accuracy and computation time follows the previous settings in Sec. 3.5.2.

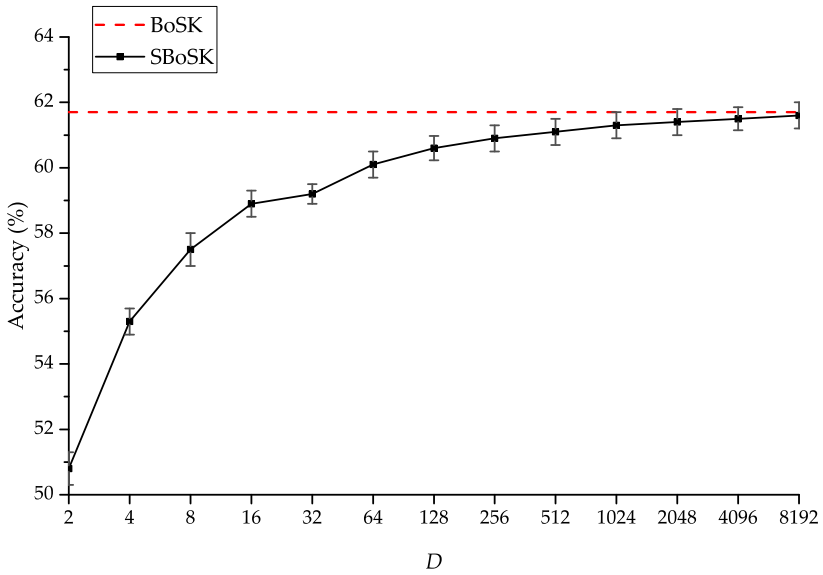


Figure 4.18: Classification accuracy comparison of BoSK and SBoSK with different dimension D (log scale). Reported accuracies and standard deviations are computed over 10 repetitions with 400 training samples per class.

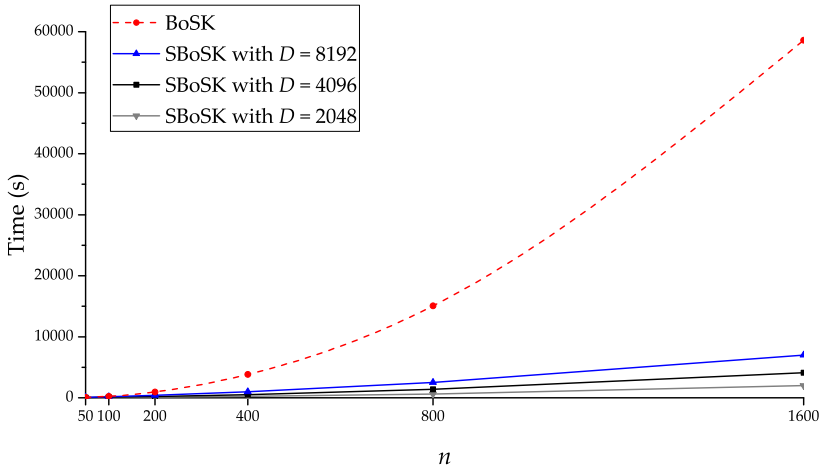


Figure 4.19: Computation time comparison of BoSK and SBoSK with $D = \{2048, 4096, 8192\}$ *w.r.t.* different number of training samples per class n

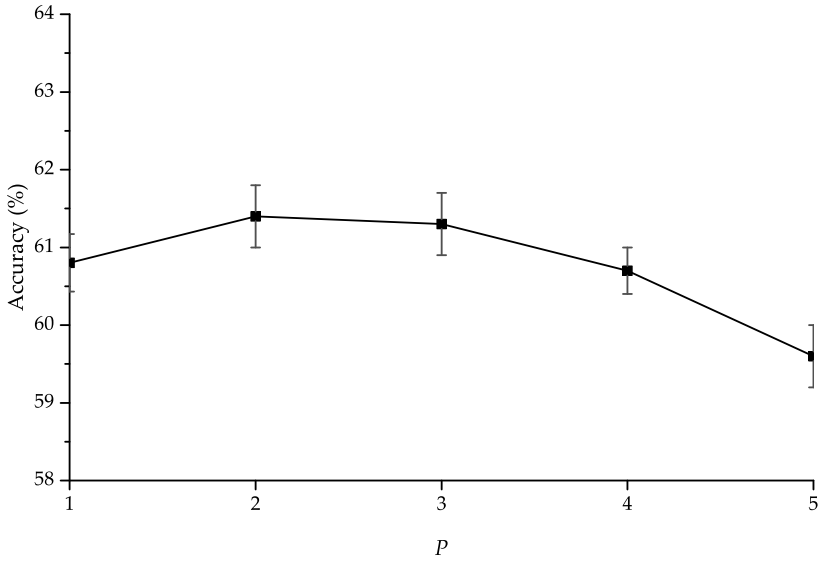


Figure 4.20: Classification accuracy *w.r.t.* different maximum considered subpath lengths P. SBoSK is computed on the VHSR image considering a kernel on trees with $D = 4096$.

As we can observe in Fig. 4.18, when RFF dimension increases, the accuracy increases till converging to the accuracy obtained with the exact computation scheme. However, such convergence rate is problem–dependent, and the number of RFF dimension is commonly set empirically [160].

Fig. 4.19 shows that the computation time increases linearly *w.r.t.* n for SBoSK, while for its exact computation, it increases quadratically. In addition, we can also observe for SBoSK that the computation time increases linearly *w.r.t.* dimension D. This calls for finding a good trade-off between the quality of approximation and the time consumption. Henceforth, in

this section, we empirically fix the RFF dimension to be $D = 4096$ as a trade-off between the approximation quality and the complexity.

As far as the maximum considered subpath P in SBoSK is concerns, we observe in Fig. 4.20 that the accuracies improve when considering subpaths with different lengths compared to using only nodes *i.e.*, $P = 1$. However, the accuracies might decrease when adding the features extracted from longer subpath patterns, calling for penalization of longer subpath.

4.5.3 Results and discussion

For the sake of comparison, we consider the Spatial Pyramid Matching (SPM) model [209], which is well known in computer vision community for taking into account the spatial relationship between a region and its subregions. The SPM relies on a quad-tree image segmentation, which split each image region iteratively into 4 square regions. In this representation, the pyramid level 0 (root) corresponds to the whole image, and the level 2 (L2) segments the image into 16 squared regions. For a fair comparison, we build SBoSK on the same spatial pyramid representation. However, let us recall that SBoSK can rely on any hierarchical representation. We thus also report the results computed on a hierarchical representation generated using Hseg. The comparison is done by randomly choosing $n = [50, 100, 200, 400]$ samples for training and the rest for testing. All reported results are computed over 10 repetitions of each run.

The classification accuracies obtained with different methods are shown in Tab. 4.1. We also provide per-class accuracies using $n = 400$ training samples in Fig. 4.21.

When compared to the Gaussian kernel computed on root regions, SBoSK consistently improves the classification results for various numbers of training samples. Furthermore, the improvements increase when more training samples are added, *i.e.* from 2.1% OA / 1.2% AA improvement with 50 training samples per class to 4.9% OA / 3.9% AA with 400 training samples per class. Analysis of the per-class accuracies leads to observing that industrial blocks and individual housing blocks, two semantically similar classes, benefit from the highest improvement among all classes. This is due to the SBoSK ability to consider object spatial decomposition and spatial relationship among its subparts.

Table 4.1: Mean (and standard deviation) of overall accuracies (OA), average accuracies (AA) and Kappa statistics (κ) computed over 10 repetitions for Strasbourg VHSR image with different training data sizes n. Best results (with a statistical significance less than 0.01% *w.r.t.* others considering the Wilcoxon signed-rank test for matched samples) are boldfaced, and numbers with * indicate that no statistically significant conclusions can be driven when compared with best results.

n		Root	SPM (L2)	SBoSK (L2)	SBoSK (Hseg)
50	OA	52.2 (0.9)	48.3 (1.8)	53.2 (1.2)	**54.3 (0.9)**
	AA	51.2 (0.7)	46.9 (1.4)	51.7 (0.4)	**52.4 (1.2)**
	κ	44.4 (0.9)	39.9 (2.0)	45.4 (1.2)	**46.6 (1.1)**
100	OA	54.2 (0.6)	50.5 (1.3)	56.0 (1.1)*	**56.5 (1.4)**
	AA	53.6 (0.4)	49.3 (0.7)	54.5 (0.7)*	**54.9 (1.1)**
	κ	46.7 (0.6)	42.3 (1.3)	48.5 (1.0)*	**49.1 (1.5)**
200	OA	55.7 (0.6)	52.4 (0.8)	57.7 (0.7)	**59.2 (0.9)**
	AA	55.1 (0.3)	51.3 (0.3)	56.5 (0.5)	**57.8 (0.9)**
	κ	48.3 (0.6)	44.5 (0.8)	50.4 (0.8)	**52.0 (1.0)**
400	OA	56.5 (0.5)	54.7 (0.5)	59.9 (0.7)	**61.4 (0.3)**
	AA	56.4 (0.2)	53.7 (0.3)	59.0 (0.6)	**60.3 (0.3)**
	κ	49.4 (0.5)	47.0 (0.5)	52.8 (0.8)	**54.4 (0.3)**

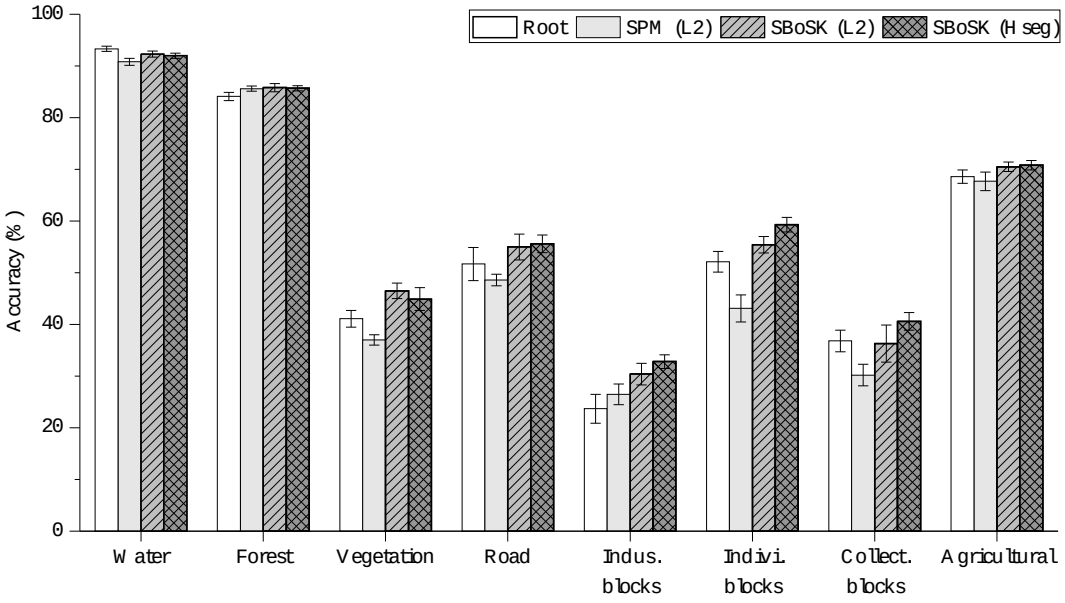


Figure 4.21: Per-class accuracies using SBoSK on spatial decomposition on Strasbourg VHSR image.

As far as the SPM model is concerned, we can see that it performs poorly with various training samples: the results drop down 3% to 4% *w.r.t.* kernel on the root region. Although SPM has been proven to be effective in computer vision domain due to its capacity of coping

with subregions and spatial arrangement betweens subregions, its one-to-one region matching strategy with exact spatial location constraint seems overstrict for remote sensing image classification. Indeed, it lacks of image orientation invariance that is required when dealing with nadir observation. To illustrate, in both individual and collective housing block classes, the orientation and absolute location of objects such as the houses in each image (40×40 pixels region) are not discriminant and thus not helpful for improving classification accuracy. However, such irrelevant features cannot be excluded in the SPM model due to its matching strategy. Therefore, two images with similar content but with different spatial locations and orientations might be classified into two different classes.

We also compare SBoSK applied on different hierarchical representations. Results show that SBoSK on Hseg segmentation leads to better results than when computed on the spatial pyramid representation. From per-class accuracies, we can see that the industrial blocks, individual housing blocks and collective housing blocks, *i.e.* semantically similar classes, are better classified. This can be easily explained by the shapes of the segmented regions: while the spatial pyramid representation splits the image into 4 squared regions independently of the actual image content, the Hseg segmentation provides a more accurate segmentation since similar regions are naturally merged together into larger regions.

4.6 Large-scale image classification on UC Merced dataset

In this section, we evaluate SBoSK on a large-scale publicly available dataset. The term large-scale refers here to large structure size (more than 300 nodes for each structured data). This number is considered as large-scale in the context of classification using structured kernel, where evaluated datasets are normally made of a few dozens of nodes each [128]. For this dataset, due to the quadratic complexity, BoSK cannot be computed, so only SBoSK is applied. RFF dimension has been empirically set to 4096.

The "UC Merced land-use" dataset (UC Merced) [213] consists of 2100 images with 256×256 pixels and 0.3-m resolution. Those images are equally distributed in 21 land-use classes, with examples from each class being shown in Fig. 4.22.

We evaluate SBoSK taking into account object spatial decomposition through hierarchical representations built on the dataset. Each image of 256×256 pixels is considered as a data instance to be classified, and is represented as a tree that can be handled with SBoSK.

In our experiment, we use two different hierarchical image representations: for the spatial pyramid representation, we define 5 levels in the pyramid that segment the image into $\{1, 4, 16, 64, 256\}$ regions. The bottom level L4 corresponds to image regions of size 16×16 pixels. For the hierarchical representation generated with Hseg segmentation, we define 5 levels of hierarchy, by empirically setting the dissimilarity criterion $\alpha = [2^5, 2^4, 2^3, 2^2]$. Such

Figure 4.22: Examples of the 21 land-use classes contained in the UC Merced dataset.

Table 4.2: Mean (and standard deviation) of overall accuracies (OA) computed over 10 repetitions and 5-fold cross validation results for UC Merced dataset with different codebook sizes and SIFT descriptors. Best results (with a statistical significance less than 0.01% *w.r.t.* others considering the Wilcoxon signed-rank test for matched samples) are boldfaced, and numbers with * indicate that no statistically significant conclusions can be driven when compared with best results.

K	Root	SPM (L2)	SPM (L4)	Spatial relatons	SBoSK (L2)	SBoSK (L4)	SBoSK (Hseg)
50	64.7 (0.7)	76.4 (0.5)	69.0 (0.3)	75.3	80.2 (0.3)	85.6 (0.3)	**87.2 (0.4)**
100	71.7 (0.4)	79.8 (0.4)	72.5 (0.4)	79.6	84.0 (0.3)	87.2 (0.3)	**88.1 (0.3)**
300	78.3 (0.3)	83.6 (0.3)	75.5 (0.3)	83.4	86.3 (0.2)	88.1 (0.3)*	**88.5 (0.3)**
500	79.8 (0.4)	84.2 (0.2)	75.9 (0.2)	85.8	87.5 (0.3)	88.7 (0.2)*	**88.7 (0.3)**
1000	81.6 (0.4)	85.1 (0.3)	75.9 (0.2)	87.6	87.9 (0.3)	88.9 (0.3)*	**88.9 (0.3)**

parameters yield a similar number of segmented regions at bottom level between both hierarchical representations, thus easing comparison between the different methods. The region feature is generated from dense SIFT descriptors with a fixed window size of 8×8 pixels and a step size of 1 pixel. It is characterized with a quantized histogram of size (also known as codebook size) $K = \{50, 100, 300, 500, 1000\}$ with K-means algorithm and Max-pooling strategy, as used in [35]. Finally, we use the Gaussian kernel computed on the square-rooted histogram [155] for each region of the SPM model and SBoSK.

All reported experiments are conducted consistently with previous evaluation procedures on this dataset [213, 35]: we randomly split the dataset to allow five-fold cross-validation and return averaged results over 10 repetitions for each randomly split dataset.

The results are shown in Tab. 4.2. We can see that SBoSK outperforms other methods for different codebook sizes, and the improvement is especially significant when the codebook

size is small.

We can see that the SPM model improves the Gaussian kernel on the root region when using two levels of pyramid (L2). However, the results drop down dramatically when four levels of pyramid (L4) are considered. This is due to the overstrict one-to-one region matching strategy adopted in SPM model (as previously discussed). On the other side, SBoSK can further improve the results when adding more pyramid representation levels from L2 to L4. This demonstrates the superiority of the proposed matching strategy relying on bags of subpaths. From the per-class accuracies shown in Fig. 4.23, we can observe the buildings, dense residential, medium residential, mobile home park, sparse residential classes, *i.e.* semantic similar classes, achieve significantly better results compared to the Gaussian kernel on the root region and the SPM model. Among these classes, the object spatial decomposition is considered as a discriminant pattern. While the SPM model fails to cope with this information, SBoSK can better incorporate this information captured from the vertical hierarchical relationships between the regions, leading to an improved accuracy. Moreover, the use of a larger hierarchical representation (*i.e.* SBoSK (L4)) leads to a further accuracy improvement for these semantically similar classes, indicating that SBoSK can benefit from better revealed object decomposition with finer details of regions and richer topological relationships among regions.

The pyramid of spatial relatons [35] is a recently proposed method tackling the issues raised when applying the SPM kernel on remote sensing images. However, we can see that SBoSK yields better results with various codebook sizes K, and the gap is significant especially when K is small. Indeed, the pyramid of spatial relatons performs similarly as the SPM kernel for 100 bins, *i.e.*, about 4% less than SBoSK using L2 and 8% less than SBoSK using L4.

Finally, when comparing SBoSK with different underlying hierarchical representations, we can notice that Hseg segmentation improves the results when the codebook size K is small. This indicates that classification results can benefit from a better hierarchical representation when region features are less discriminant. Since the object decomposition are better revealed with Hseg segmentation, we claim that such topological features are especially useful when the region appearance feature is not discriminative enough.

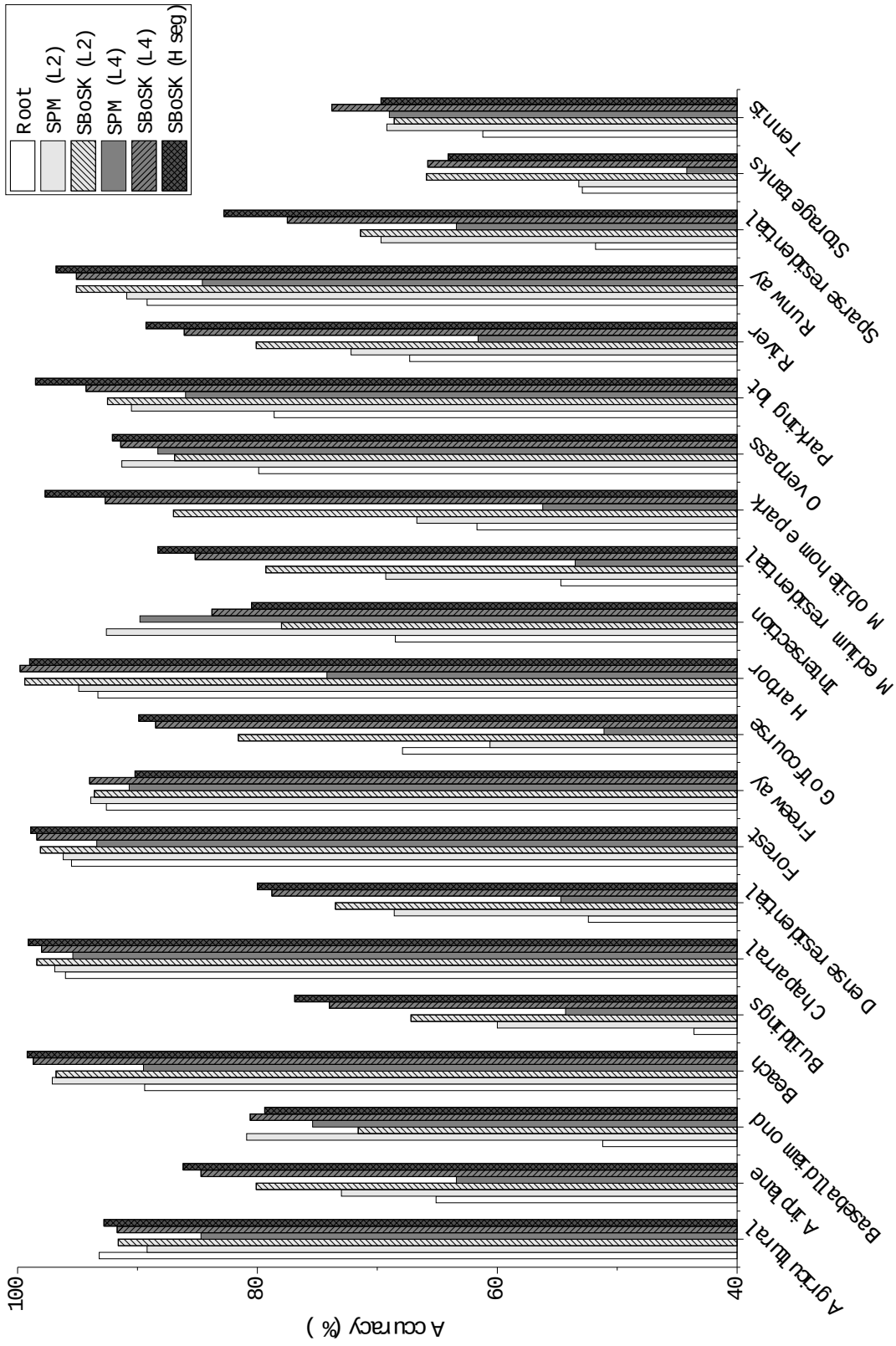


Figure 4.23: Per-class accuracies with spatial decomposition-based classification on UC Merced dataset with codebook size $K = 100$.

4.7 Chapter summary

An image classification approach based on object spatial decomposition is proposed in this chapter. The decomposition can be expressed as a tree structure $\mathcal{T}$, where objects-of-interest are the nodes of the tree and the hierarchical relationships among them are modeled through the edges. We thus suggest to apply (S)BoSK on the induced tree structures. The proposed (S)BoSK working on hierarchical representations is closely related to the Spatial Pyramid Matching (SPM) kernel, one of the most standard way to consider object spatial decomposition. However, (S)BoSK can be applied for arbitrary hierarchical representations, and is robust to image rotation and translation.

A rigorous experimental plan based on an artificial dataset is used to evaluate (S)BoSK, in which we analyzed the kernel with different weighting schemes, as well as SBoSK approximation results using Random Fourier Features. Results indicate that BoSK can maintain a good performance up to a certain ratio of structure distortion, among which the weighting strategy imposing a maximum length of subpaths yields a better results compared to other weighting strategies. The analysis of SBoSK confirms that it gives similar results than BoSK, both in term of Gram matrix similarity and classification accuracies. Moreover, SBoSK has clear advantages in the context of large tree sizes and large number of training data size.

We also carried out experiments on two remote sensing datasets, including one publicly accessible dataset: UC Merced dataset. Results confirmed the effectiveness of (S)BoSK for taking into account object spatial decomposition. Comparing to related state-of-the-art methods, including SPM and its variant dedicated to geographic images, we observed a clear gain in terms of classification accuracy.

Chapter 5

Multi-source and multi-resolution image classification

Contents

In the previous two chapters, we have illustrated how (S)BoSK kernel was able to take into account the contextual and the spatial decomposition information, respectively. Meanwhile, we have classified so far only the leaves (corresponding to the pixels in the image) or the roots (corresponding to the image tiles) separately.

We propose in this chapter a novel multi-source and multi-resolution image classification method. It relies on the combination of (S)BoSK operating on a hierarchical image representation built from multi-source and multi-resolution images, allowing one to benefit from both contextual and spatial decomposition information simultaneously.

5.1 Introduction

Data fusion approaches have gained increasing interest recently in the remote sensing community [78, 114, 119], thanks to the recent technologies that make multiple and heterogeneous image sources available for the same geographical Earth surface area. Facing the challenges of the acquisition of large amounts of different resolutions (spatial, spectral, temporal) images and heterogeneous sources (Optical, SAR, LiDAR), data fusion techniques have demonstrated the interests of exploiting complementary information of the observed scene carried with different imaging modalities. For instance, combining high-resolution imagery and LIDAR data allows better accuracy achievements in a urban area classification task [36, 52]. As the availability of multi-resolution remote sensing data is rapidly increasing, methods able to fuse data from multiple sources and at multiple resolutions are becoming an important research topic in remote sensing [218, 78].

Meanwhile, hierarchical image representations are becoming more and more popular in the remote sensing community thanks to their capability of revealing objects-of-interest at various scales and modeling their topological relationships [17]. Their use for multi-source and multi-resolution image classification remains however to be demonstrated and is the main objective of this chapter.

We propose here a novel multi-source and multi-resolution classification approach relying on (S)BoSK and operating on a hierarchical image representation built from two images at different resolutions, possibly with different modalities. Both images capture the same scene with different sensors and are joined together through the hierarchical representation, where, for instance, coarser levels are built from a Low Spatial Resolution (LSR) or Medium Spatial Resolution (MSR) image while finer levels are generated from a High Spatial Resolution (HSR) or Very High Spatial Resolution (VHSR) image. Therefore, we assume an integer scale ratio between the resolutions of LSR/MSR and of HSR/VHSR, requiring both images to be perfectly overlapping. In addition, as two images at different resolutions provide finer and coarser levels of the hierarchical representation respectively, their resolution must differ enough to allow complementary viewpoints over the same area.

Building the hierarchical representation of two images at different resolutions allows one to benefit from the contextual information thanks to the coarser levels, and from the object spatial decomposition thanks to the finer levels. Two (S)BoSK are then used to perform machine learning directly on the constructed hierarchical representation and are combined together. This strategy overcomes the limits of conventional remote sensing image classification procedures that can handle only one or very few pre-selected scales of hierarchical representation. Experiments run on a urban classification task show that the proposed approach can highly improve the classification accuracy *w.r.t.* conventional approaches working on a single scale.

The chapter is organized as follows: a brief review of related work is provided in Sec. 5.2. We then describe in Sec. 5.3 the proposed multisource classification method, which is followed by a concrete example and its evaluation in Sec. 5.4. The chapter ends with conclusion and discussion in Sec. 5.5.

5.2 Related work

5.2.1 Data fusion in remote sensing

Data fusion aims to combine data from various sources and to provide more detailed information. This covers a large range of applications and research directions, *e.g.* image fusion such as pan–sharpening approaches [129], image classification using multisource data [194], and multiangular, multitemporal image analysis [149]. Here we concentrate on the techniques that allow fusing data from multiple sources and captured at multiple resolutions. As each sensor provides some unique spatial details of the observed scene, exploring and combining such information is important. Methods that can fuse multi-source and multi-resolution data have been proved to be effective for improving classification accuracy [218]. Among famous remote sensing fusion strategies, two main directions can be found in literature: decision level fusion and feature level fusion.

Fusion at the decision level involves mostly defining a strategy to combine results obtained with multiple classifiers. In general, each data source is classified separately and the classification outputs are fused together to produce the final classification map. For instance, [60] propose to use a set of SVM classifiers, with each of them classifies one data source separately, and the final class label is chosen with a majority voting scheme. In [205], the output values in the decision function of SVM classifiers (learned on each data source) are trained again with another SVM classifier in order to determine the final class label. Other approaches consisting of combining multiple classifiers, *e.g.* bagging and boosting strategies, have been evaluated [24, 12] in the context of classification using multi-source remote sensing images.

Fusion can also be achieved at the feature level. Among popular techniques, feature vector extraction with different image sources and concatenation of the extracted vectors can be considered as the most straightforward way to combine multi-source information [191, 189, 78]. Other strategies such as including different data sources directly inside the classification methods have also been used. In [195], relevant features of various modalities are learned and combined into a single classifier. These features are added by iteratively checking whether including them can maximize the separating margin in SVM. In [174, 135], different sources of information are represented as separate feature vectors, and are joined together in a probabilistic Markov model in the unary energy term through combining their

conditional probabilities *w.r.t.* related class labels.

Among techniques that are capable of performing data fusion at feature level, kernel methods have been identified as one of the most studied research directions in a recent survey paper [78], as they offer a general framework allowing one to fuse different sources of information easily in a classification problem. In this framework, kernels are computed from different data sources and all the source-specific kernel matrices are combined into a final one before using kernel-based classification methods. Such techniques have been used for combining spectral and spatial information extracted from multi-source remote sensing images [194], as well as multi-temporal remote sensing images [28].

In [194, 28], kernels are computed from different image sources and fused through a linear kernel combination before using a SVM. Their importance is coped with a weighting parameter and is determined by cross-validatition. Such weights can also be learned through a multiple kernel learning framework [120, 79], which is especially useful when the number of kernels increases and weighting parameters become hard to tune [188].

5.2.2 Fusion with multiple spatial resolution images

The aforementioned data fusion methods have been categorized from a theoretical point of view. We pay here a special attention to methods that able to exploit multiple spatial resolution images from the same geographical area.

In [176], two different resolution images are used. The higher resolution image provides fine details of image content, on which class labels are defined, while lower resolution allows exploiting richer spectral information. Two complementary information are modeled within a Bayesian framework, to affine the prediction and to produce a more accurate classification map.

In [107], a hierarchical representation is built with different resolution images. The representation is constructed iteratively, where, at each step, the segmentation map obtained with the lower resolution image is used as an input for the higher resolution image in order to generate a finer level segmentation. For extracting features of regions in a hierarchy using multiresolution images, [206, 107] propose to characterize the regions at coarser level with the histogram constructed from the pixel spectral information in the higher resolution image.

In [90, 91, 203], multiresolution images are used to build a quad-tree (pyramid) representation, where the bottom of pyramid is set to the panchromatic image of higher resolution and top level is associated to the multi-spectral image of lower resolution. A hierarchical MRF model is further built on the constructed quad-tree representation to model the parent-children relationships among pixels at two different resolution images.

In this chapter, we also propose to build a hierarchical image representation using different resolution images. However, the representation is mainly dedicated to reveal the topological information among objects at various scales *i.e.* contextual and object decomposition information. In order to take into account these two types of information, we fuse two (S)BoSK that are computed from different images through linear kernel combination, with an additional parameter controlling the importance of each kernel. The fusion of two SBoSK can also be viewed as feature vector concatenation, since RFF embedding of SBoSK yields a vector form.

5.3 Multi-source images classification

We introduce a novel approach i) to build a hierarchical image representation from a pair of images with different resolutions (captured with two different sensors), and ii) to combine two (S)BoSK to perform supervised classification directly from the constructed tree.

5.3.1 Building the hierarchical representation

We join two resolution images into a single hierarchical representation through two separate steps: i) use a LSR/MSR image to construct the coarser levels of the hierarchy where contextual information can be captured on the one side, ii) use a HSR/VHSR image to generate the finer levels of the tree, where object spatial decomposition are modeled on the other side.

To be more specific, we firstly initialize the segmentation at the pixel level on the LSR/MSR image and construct the coarser levels. Let n_1 be a data instance to be classified. Within the LSR/MSR image, it corresponds to a pixel n_1^l and can be featured as a path $\mathcal{P} = \{n_1^l, ..., n_P^l\}$ that models the evolution of the pixel n_1^l through the hierarchy. Each node n_i^l is described by a d-dimensional feature $\boldsymbol{x}_{n_i^l}$ that encodes the region characteristics, *e.g.* spectral information, size, shape, etc.

Secondly, we use the HSR/VHSR image to provide the fine details of the observed scene for each data instance n_1. Due to the pixel resolution difference, one pixel of the LSR/MSR image n_1^l corresponds to a region of the HSR/VHSR image n_1^h. Therefore, we initialize the top level of the multiscale segmentation to be the corresponding regions, then construct the finer levels. Through the hierarchy, the data instance n_1 can be modeled as a tree $\mathcal{T}$ rooted in n_1^h which encodes object decomposition and the topological relationships among its subparts. The characteristics of a region n_i^h are also described by a feature vector $\boldsymbol{x}_{n_i^h}$. Note that $\boldsymbol{x}_{n_i^l}$ and $\boldsymbol{x}_{n_i^h}$ can be extracted with different modalities, thus the features can also have different dimensions.

In the end, each data instance n_1 can be represented by an ascending path $\mathcal{P}$ from the LSR/MSR image, and a descending tree $\mathcal{T}$ generated from the HSR/VHSR image.

5.3.2 Fusion of (S)BoSK

To perform image classification from a hierarchical representation, we propose to combine two (S)BoSK computed on paths $\mathcal{P}$ and trees $\mathcal{T}$ respectively. Both (S)BoSK exploit complementary information from the hierarchical representation, therefore they are combined at the end through a kernel combination step. The final kernel between two data instances $K(n_1, n_1')$ is computed using a linear combination of the two (S)BoSK:

$$\begin{aligned} K(n_1, n_1') &= \rho \times K(\mathcal{P}, \mathcal{P}') + (1-\rho) \times K(\mathcal{T}, \mathcal{T}') \\ &= \rho \times \tau(\boldsymbol{s} \in \mathcal{P})^T \tau(\boldsymbol{s}' \in \mathcal{P}') + (1-\rho) \times \tau(\boldsymbol{s} \in \mathcal{T})^T \tau(\boldsymbol{s}' \in \mathcal{T}') \\ &= \left[\sqrt{\rho} \times \tau(\boldsymbol{s} \in \mathcal{P})^T, \sqrt{1-\rho} \times \tau(\boldsymbol{s} \in \mathcal{T})^T\right]^T \left[\sqrt{\rho} \times \tau(\boldsymbol{s}' \in \mathcal{P}')^T, \sqrt{1-\rho} \times \tau(\boldsymbol{s}' \in \mathcal{T}')^T\right], \end{aligned} \tag{5.1}$$

where $K(\mathcal{P}, \mathcal{P}')$ is BoSK on paths, and $K(\mathcal{T}, \mathcal{T}')$ is BoSK on trees, $\tau(\boldsymbol{s} \in \mathcal{P})$ and $\tau(\boldsymbol{s} \in \mathcal{T})$ are RFF embedding of $\mathcal{P}$ and $\mathcal{T}$, respectively, and according to Algorithm 1, with a parameter $\rho \in [0, 1]$ that controls the importance ratio between the two kernels. Such an embedding allows computing the fused kernel through inner product of concatenated feature vectors. It computes each data instance independently, yielding a linear complexity *w.r.t.* training sample size and maintaining the overall scalability of the proposed classification approach.

In the following section, we show a concrete example of the proposed multi-source image classification method using MSR and VHSR remote sensing images.

5.4 Evaluation on Strasbourg dataset using both Spot-4 and Pleaides images

In this section, we evaluate the proposed approach focusing on urban land-use classification in the South of Strasbourg city, France. Two images are considered, both capturing the same geographical area with different sources:

- a MSR image (previously introduced in Sec. 3.5.1), captured by a Spot-4 sensor, containing 326×135 pixels at a 20 m spatial resolution, described by 4 spectral bands: Green, Red, NIR, MIR.

- a VHSR image (previously introduced in Sec. 4.5.1), captured by a Pleiades satellite, containing 13040×5400 pixels at a 0.5 m spatial resolution (obtained with pan-sharpening technique), described by 4 spectral bands: Red, Green, Blue, NIR.

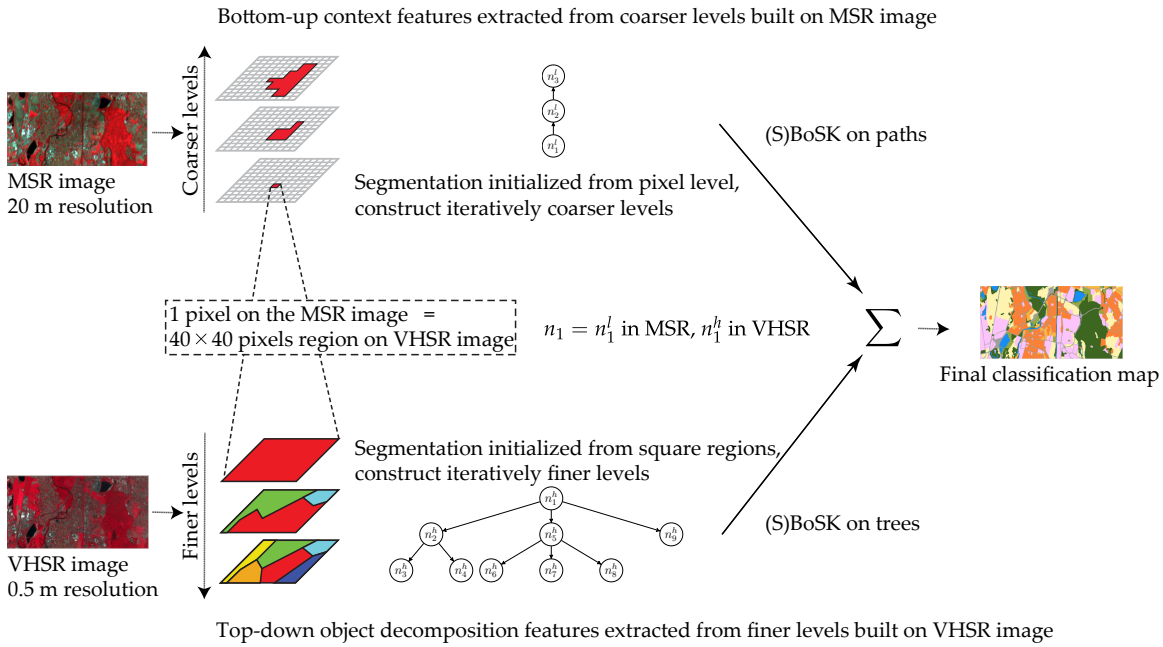


Figure 5.1: Illustration of the hierarchical image representation for one data instance n_1, and data fusion with (S)BoSK. Each data instance corresponds to a pixel of the MSR image n_1^l, and a 40×40 square region on the VHSR image n_1^h. It associates the contextual information thanks to the coarser levels of the hierarchy built from the MSR image, and the object spatial decomposition information thanks to the finer levels constructed on the VHSR image. Both complementary information are taken into consideration thanks to two dedicated structured kernels, then fused together providing the final classification output.

Let us note that while the two images have here the same number of spectral bands, it is not a required condition of our algorithm that is able to cope with very different image types.

Each pixel on the MSR image corresponds to a square region of size 40×40 pixels on the VHSR image. Both image are joined together through a hierarchical representation, as illustrated in Fig 5.1.

For a comparison purpose, the following scenarios are considered:

- scenario 1: Gaussian kernel at single level on the MSR image *vs.* SBoSK taking into account the contextual information at multiple levels on the MSR image. Recall that a detailed analysis has been done in Sec. 3.5.3.

- scenario 2: Gaussian kernel at single level on the VHSR image *vs.* SBoSK taking into account the object spatial decomposition at multiple levels on the VHSR image. Recall that a detailed analysis has been done in Sec. 4.5.3.

- scenario 3: combining both the contextual and object spatial decomposition information modeled through a hierarchical representation using both MSR and VHSR images.

The classification accuracies achieved with the different methods are shown in Tab. 5.1 using various numbers of training samples $n = [50, 100, 200, 400]$. We also show the per-class accuracies for the 8 different classes using $n = 400$ training samples in Fig. 5.3.

The classification results show that combining contextual and decomposition information leads to a significant improvement. Indeed we observe, for various training sample sizes, more than 4% improvement over SBoSK on a single MSR image, and more than 10% improvement over SBoSK on a single VHSR image. From an analysis of per-class accuracies achieved with SBoSK, we can see that some classes (urban vegetation, industrial blocks, individual and collective housing blocks and agricultural zones) yield higher accuracies on the MSR image, while some other classes (water surfaces, forest areas, roads) obtained better accuracies on the VHSR image. Nevertheless, combining both kernels allows benefiting from the advantages of the two types of complementary information, thus yielding to the best accuracies for all classes. Indeed, we can state that the prediction achieves a spatial regularization for the large regions (*e.g.* industrial and individual housing blocks) thanks to the contextual information, while providing precision for the small structures (such as road networks) thanks to the detailed object spatial decomposition information.

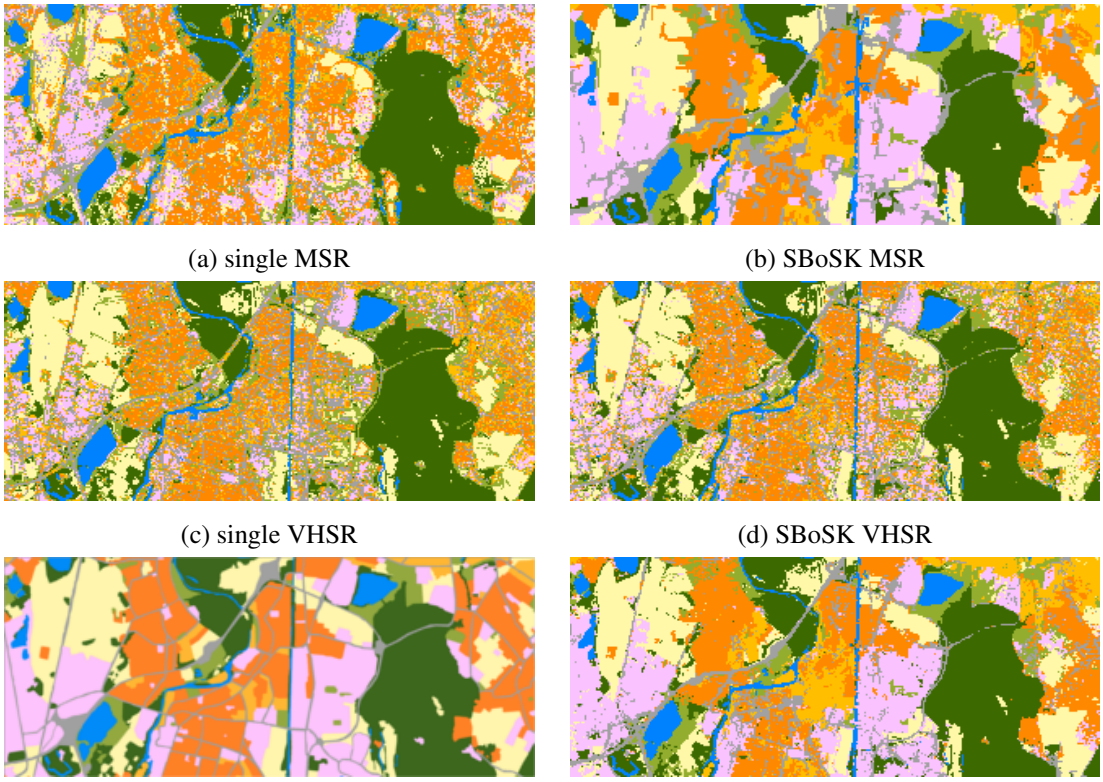
When compared with the Gaussian kernel computed on a single image at single level, combining both SBoSK built upon two different image sources achieves 13% OA improvement when using $n = 50$ and 20% OA improvement when using $n = 400$. This demonstrates the superiority of our proposed multi-source classification method that is able to exploit topological information across multiple scales.

As shown in Fig. 5.2a, the predictions are very noisy with a single level analysis of the MSR image. This is the typical "salt and pepper" problem encountered in remote sensing image classification when the spatial information is not taken into account. Using multi-scale information, the spatial dimension is implicitly taken into consideration by the ancestor regions in the hierarchy. Thus a smoother prediction map can be obtained (as shown in Fig. 5.2b). Let us note that we did not use any post-processing technique to produce such a classification map, relying only a structured kernel coping with context information. However, we can also observe that small structures such as road networks disappear in certain areas, and enhance wrongly in other ones.

As far as the VHSR image is concerned, using SBoSK leads to a more precise prediction for most of classes when compared to a single level analysis of the VHSR image (as shown in Fig. 5.2c and Fig. 5.2d). However, the prediction maps are noisy with both single and multiple scales.

Combing both SBoSK computed on MSR and VHSR image manages to benefit from the

advantages brought by the two complementary information sources. We can see in Fig. 5.2f that the prediction seems to achieve a spatial regularization for the large regions, while providing precision for the small structures such as road networks.



(a) single MSR (b) SBoSK MSR

(c) single VHSR (d) SBoSK VHSR

(e) Ground truth image (f) Combined

Figure 5.2: Classification maps for methods using single and multiple levels of a hierarchical image representation. Scenario 1: single level on Spot-4 image **(a)** *vs.* multiple levels contextual information on Spot-4 image **(b)**; scenario 2: single level on Pleiades image **(c)** *vs.* multiple levels spatial decomposition information on Pleiades image **(d)**; scenario 3: combination of contextual and spatial decomposition information **(f)**. Ground truth image **(e)** is also given as reference.

Table 5.1: Mean (and standard deviation) of overall accuracies (OA), average accuracies (AA) and Kappa statistics (κ) computed over 10 repetitions for Strasbourg MSR and VHSR images with different training data sizes n. Best results (with a statistical significance less than 0.01% *w.r.t.* others considering the Wilcoxon signed-rank test for matched samples) are boldfaced.

n		Single MSR	SBoSK MSR	Single VHSR	SBoSK VHSR	Combined
50	OA	45.3 (2.3)	57.8 (1.3)	52.2 (0.9)	54.3 (0.9)	**65.3 (0.6)**
	AA	43.9 (1.0)	57.9 (0.8)	51.2 (0.7)	52.4 (1.2)	**64.3 (0.8)**
	κ	32.2 (2.1)	50.2 (1.4)	44.4 (0.9)	46.6 (1.1)	**58.9 (0.7)**
100	OA	47.9 (1.3)	63.3 (0.7)	54.2 (0.6)	56.5 (1.4)	**69.8 (0.7)**
	AA	46.2 (0.5)	64.0 (0.7)	53.6 (0.4)	54.9 (1.1)	**69.8 (0.8)**
	κ	39.1 (1.3)	56.5 (0.8)	46.7 (0.6)	49.1 (1.5)	**64.1 (0.8)**
200	OA	51.4 (0.8)	68.4 (0.7)	55.7 (0.6)	59.2 (0.9)	**73.9 (0.5)**
	AA	48.1 (0.4)	69.7 (0.5)	55.1 (0.3)	57.8 (0.9)	**74.8 (0.3)**
	κ	42.6 (0.8)	62.3 (0.7)	48.3 (0.6)	52.0 (1.0)	**68.7 (0.3)**
400	OA	52.2 (0.4)	73.0 (0.4)	56.5 (0.5)	61.4 (0.3)	**77.3 (0.3)**
	AA	49.1 (0.2)	74.8 (0.4)	56.4 (0.2)	60.3 (0.3)	**79.1 (0.4)**
	κ	43.5 (0.4)	67.6 (0.5)	49.4 (0.5)	54.4 (0.3)	**72.7 (0.4)**

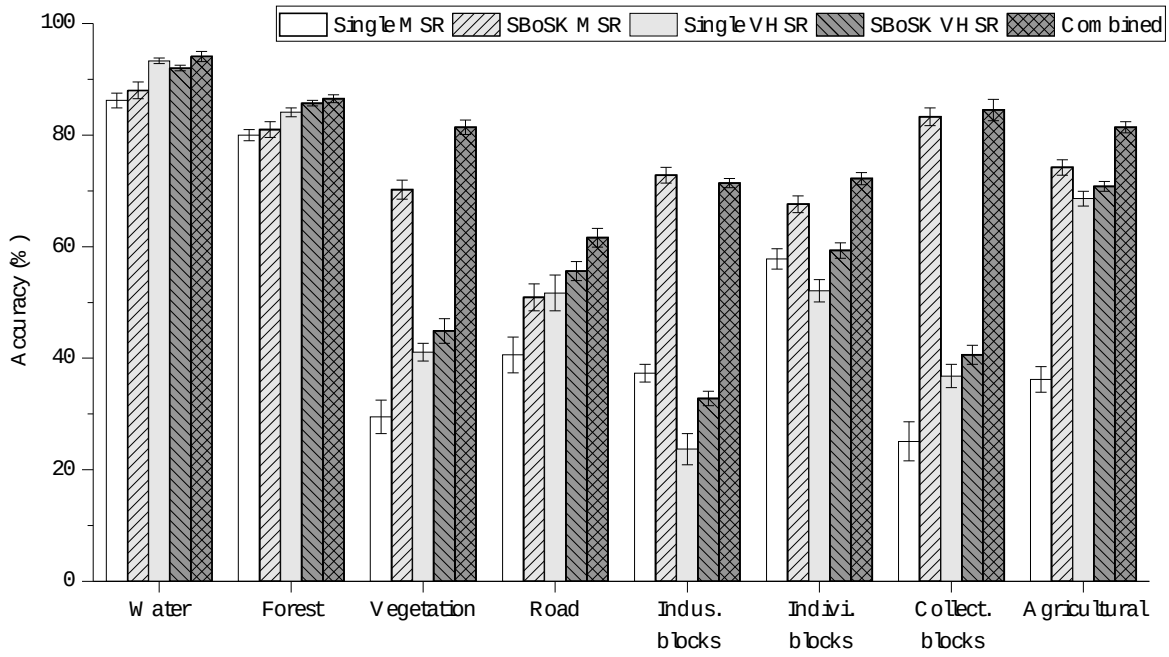


Figure 5.3: Per-class accuracies for multi-source classification using Strasbourg MSR and VHSR images.

5.5 Chapter summary

A novel multi-source and multi-resolution image classification method is presented in this chapter. The proposed method joins two resolution images taken from the same area through a hierarchical representation. In such a representation, we consider each pixel at lower resolution image as the data instance to be classified. Due to the overlapping of the two images, the data instance also corresponds to a region when projecting it on the higher resolution image. To build the hierarchical representation, we rely on two steps: from the lower resolution image are constructed the coarser levels of hierarchy where contextual information of each data instance can be revealed; from the higher resolution image are generated the finer levels of the tree, where spatial decomposition of data instances are modeled. Two (S)BoSK are then used to perform machine learning directly on the constructed hierarchical representation, aiming at combining both contextual and decomposition information into a a unique, multi-source and multi-resolution classification scheme.

This strategy overcomes the limits of conventional remote sensing image classification procedures that can handle only one or very few pre-selected scales of hierarchical representation. Experiments run on a urban classification task show that the proposed combination of two (S)BoSK can highly improve the classification accuracy compared to constructing kernel on a single scale and on a single source.

Chapter 6

Conclusions and perspectives

Contents

This final chapter aims to conclude the manuscript. We first summarize our main contributions arising from this work in Sec. 6.1. Then in Sec. 6.2, we highlight several directions for future works, which can potentially improve the methods proposed in this thesis, as well as open new perspectives for machine learning on hierarchical image representations.

6.1 Conclusions

In this thesis, we addressed image classification problems using hierarchical image representations. These hierarchical representations can reveal objects-of-interest at different scales, as well as their topological relationships across the scales. Their successful applications in the spatial-spectral pixel-wise remote sensing image classification and multiscale object-based image analysis frameworks motivate us to develop novel approaches relying on kernel-based machine learning techniques, in order to fully exploit the topological features among objects that are provided by hierarchical representations.

We first begin by presenting our context in Chap. 1. This chapter offers a general presentation of the remote sensing classification scheme, its related challenges, as well as a general introduction about our main context: when kernel-based machine learning meets hierarchical image representations.

Our main contributions consist of designing a new structured kernel and using it to solve various remote sensing image classification problems.

Chap. 2 introduces our proposed kernel relying on subpath substructures under the convolution kernel framework, called Bag of Subpaths kernel (BoSK). It can be applied for unordered tree as well as path structured data equipped with numerical features, capturing the vertical hierarchical relationships among nodes in the structured data. An efficient iterative algorithm is proposed for exact kernel computation that calculates pairwise BoSK in a quadratic complexity *w.r.t.* structure size. This algorithm is efficient for small structures and small training data size. We also proposed its scalable version, Scalable Bag of Subpaths kernel (SBoSK), based on applying Random Fourier Features for atomic kernel (*i.e.* the Gaussian kernel) approximation. Such technique maps the structured data in a randomized finite-dimensional Euclidean space, where inner product of the transformed feature vector approximates BoSK. It brings down the complexity from quadratic to linear *w.r.t.* structure size and *w.r.t.* volumes of data, making the kernel relevant even in a large-scale machine learning context.

Following the introduction of our structured kernel (S)BoSK, we presented its first application in Chap. 3. We took into account the contextual information for pixel-wise classification, as the context of each pixel can be modeled as a path structure extracted from a hierarchical representation. In Chap. 4, we presented the second application of (S)BoSK, whose goal is to rely on the spatial decomposition for sub-image/tile-based image classification. Indeed, such a decomposition information can be extracted from hierarchical representation and modeled as a tree structure. Evaluations on various datasets, including several publicly available ones, indicate the superiority of (S)BoSK *w.r.t.* respective state-of-the-art methods in both scenarios.

After confirming that (S)BoSK can benefit from either contextual or decomposition infor-

mation extracted from hierarchical image representations, we proposed a novel multiscale classification approach in Chap. 5. It operates on a hierarchical image representation built from two images provided with different spatial resolutions. Both images capture the same scene but with different sensors, and thus can be naturally combined together through a unique hierarchical representation. In such a representation, coarser levels are built from a Low Spatial Resolution (LSR) or Medium Spatial Resolution (MSR) image, while finer levels are generated from a High Spatial Resolution (HSR) or Very High Spatial Resolution (VHSR) image. One can thus benefit from the contextual information thanks to the coarser levels, and spatial decomposition information thanks to the finer levels. Two dedicated (S)BoSK are then used to perform machine learning directly on the combined hierarchical representation. This strategy overcomes the limits of conventional remote sensing image classification procedures that can handle only one or very few pre-selected scales. Experiments run on an urban classification task showed that the proposed approach can highly improve the classification accuracy *w.r.t.* conventional approaches working on a single scale.

6.2 Perspectives

In this section, we propose several interesting directions as a continuation of this thesis. We first introduce new strategies for improving the proposed methods by addressing various aspects of proposed (S)BoSK, then we offer a selection of possible future directions in machine learning on hierarchical image representation.

6.2.1 Improvements of the proposed methods

As the (S)BoSK is the key part for learning on hierarchical image representation, improving (S)BoSK can have direct impact on the classification accuracy using either contextual information presented in Chap. 3, or spatial decomposition information introduced in Chap. 4, and even both through our proposed multiscale classification approach in Chap. 5.

Multiple Kernel Learning

The weighting strategies adopted in (S)BoSK are inspired from the literature on structured kernels, while advanced weighting strategies, such as Multiple Kernel Learning [79] could have been explored. (S)BoSK is built on a linear combination of weighted individual kernels, where each individual kernel is computed using one specific subpath length. Therefore, we can deploy an automatic procedure based on MKL to weight the different subpath lengths. An improvement of classification performances could be expected by the optimal weight computation.

The multiple Kernel Learning framework has been proved to be effective for kernel methods. Its scalability is limited by kernel matrix computation, while our SBoSK is designed for large-scale datasets. In order to maintain the scalability, vector-based group feature learning model (*e.g.* group lasso regularization [189]) can be applied for learning optimal weight for each subpath length. The embedded vector (*i.e.* explicit feature map using Random Fourier Features) for each subpath length can be considered as one feature group, the learned weights can be taken into account through factorization of embedded vector, without any change of the SBoSK algorithm.

Learning the discriminative subpath patterns

The aforementioned weighting scheme can be applied for computing an optimal weight for each subpath length. However inside each bag of specific length, the kernel values sum up together without any weighting scheme.

In fact, this is commonly used in the convolution kernel framework, while some studies argue that the direct sum operation inside such framework might not be effective [57, 178]: if the bags contain lots of substructures, the kernel tends to average the information, and the discriminative power might thus be reduced.

Meanwhile, strategies using a weighted sum for different elements within each bags have been largely used in the computer vision community. These strategies are proposed either for reducing the negative effects of frequent elements (*e.g.* reduce burstiness effect [186]; balance the influence of frequent and rare descriptors [141, 103]), or for increasing the contributions of the determinative patterns through a higher weight [82].

These weighting strategies could be useful for (S)BoSK. Instead of weighting all subpaths constantly during aggregation (sum over all matched pairs of subpaths), the more discriminative ones could be associated with higher weights.

Learning kernel approximation

Currently, the proposed SBoSK can only be applied in case of Gaussian atomic kernel, due to the Random Fourier Features, and theoretical issues derived from convolution kernel framework. Although Gaussian kernel is largely used, other kernels are relevant for different feature vector representations, *e.g.* χ^2 kernel has been proved to be effective in case of histograms.

Recently, the explicit maps for approximating different well-known kernels have been analyzed in the large-scale machine learning context [117, 116, 198]. Moreover, strategies able to learn arbitrary kernel approximations have been investigated in the literature [171].

Therefore, application of the proposed SBoSK to other atomic kernels are sought and using pre-existing approximation maps or learning kernel approximations could be one solution to be explored.

Complexity

From a complexity point of view, dimension reduction techniques can be applied on the resulting RFF embedded vector in order to further decrease the computation time.

As illustrated in the thesis, the RFF dimension is highly related to the kernel approximation error: the higher the RFF dimension is, the better kernel approximation can be achieved. Therefore, the embedded vector might have thousands of dimensions. It not only slows down the training time and prediction time, but also leads to a higher storage footprint.

In fact, the same issue occurs in the computer vision community when using "Bag of visual Words" framework: the number of visual words is normally more than one thousand, or even ten thousands, for maintaining a good performance [33]. Dimension reduction techniques, especially Principal Component Analysis (PCA), are successfully applied for post-processing high-dimensional descriptors faced in the community [54, 154]. Such techniques could be applied after SBoSK embedding in order to further decrease the computation time.

Effect of different hierarchical representations

In this thesis, we have provided only a limited study of the effect of the underlying hierarchical representation on the overall classification result. Although it has been shown that a better hierarchical representation can result in improving the classification accuracy, such conclusion remains fairly intuitive. A better understanding of relationships between effectiveness of (S)BoSK and underlying hierarchical representations would be one of our next steps.

6.2.2 A step further

Multiscale image analysis

We have shown through this thesis the powerfulness of image classification incorporating multiscale topological information revealed through hierarchical representations. However, the data instances to be classified in our proposed applications are always relying on one single specific scale, *i.e.* pixels or tree leaves in Chap. 3, tiles or tree roots in Chap. 4, or one specific intermediary level in Chap. 5. Our final goal is to classify each region in the hierarchical representation using its contextual and decomposition information, to perform a full multiscale image analysis, where each node of the tree is given a semantic label.

In order to allow such a multiscale analysis, our proposed SBoSK can be used as a region kernel descriptor [18], through Random Fourier Features embedding. The embedded vector for each region describes both the contextual and decomposition information in a fixed size dimension, as we proposed in Chap. 2. With such kernel descriptors for each region, we can benefit a large number of machine learning techniques that can predict consistent labels in trees. For instance, graphical models such as in [7, 162] employ probabilistic inference techniques that allow labeling the nodes at multiple scales in a hierarchical representation (tree-structured graphical models), increasing the chance to estimate accurately the object boundaries.

In addition, in the context of multiscale image analysis, the class labels are often defined at multiple scales and have certain hierarchical relations among classes. For instance, building and tree species at finer level can form residential blocks at intermediary level, and groups of residential blocks can form urban area at coarse scale. Such definition is especially popular in the GEOBIA community [158] and is naturally revealed in a hierarchical image representation. A possible extension of this work to multiscale analysis could be done through exploration of structured output learning framework [146]

Other machine learning frameworks

The proposed (S)BoSK is evaluated in a supervised learning context (mainly SVM) for remote sensing image classification, while its application can be extended to other kernel methods *e.g.* kernel discriminant analysis [181], or even in unsupervised learning frameworks *e.g.* kernel clustering [64]. More recently, kernel clustering using Random Fourier Features [38] has been proposed for capturing the non-linear patterns while maintaining its scalability for large dataset. Following this direction, our proposed SBoSK can be directly extended to perform remote sensing clustering (taken into account the spatial domain) in a large-scale context.

Another popular unsupervised learning framework in remote sensing community is Kernel Principal Component Analysis (KPCA). It can be used for target detection and anomaly detection [32, 108]. Our proposed (S)BoSK can be extended for these tasks. In addition, recent kernel approximation techniques allow large-scale applications *e.g.* Randomized Nonlinear PCA [122], which is also highly related to SBoSK.

Having proven the relevance of kernel-based learning of hierarchical image representations in the supervised case, *i.e.* a conventional paradigm, we can now explore more recent paradigms from machine learning. Among them, active learning allows selecting the most useful samples from unlabeled ones, and adding them into the training set to improve the discrimination capabilities of the model. It is especially useful in real world remote sensing image classification [193], where collecting the training samples is costly. As the spatial do-

main has been implicitly taken into account by the proposed SBoSK, the selected samples considered in the active learning framework are expected to be spatially well distributed. Some further improvements of classification results can thus be expected [151].

Domain adaptation have become popular recently for real world remote sensing image classification, as the collected training samples might be different from those needed to be predicted [192]. As such a problem also exists when performing machine learning from hierarchical image representations, exploring the techniques for domain adaptation could be a future direction.

Bibliography

[1] Fabio Aiolli, Giovanni Da San Martino, and Alessandro Sperduti. “An efficient topological distance-based tree kernel”. In: *IEEE Transactions on Neural Networks and Learning Systems* 26.5 (2015), pp. 1115–1120.

[2] Fabio Aiolli, Giovanni Da San Martino, and Alessandro Sperduti. “Route kernels for trees”. In: *International Conference on Machine Learning*. 2009, pp. 17–24.

[3] Antti Airola, Sampo Pyysalo, Jari Björne, Tapio Pahikkala, Filip Ginter, and Tapio Salakoski. “All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning”. In: *BMC bioinformatics* 9.11 (2008), S2–60.

[4] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. “Good practice in large-scale learning for image classification”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.3 (2014), pp. 507–520.

[5] Selim Aksoy and R Gökberk Cinbis. “Image mining using directional spatial constraints”. In: *IEEE Geoscience and Remote Sensing Letters* 7.1 (2010), pp. 33–37.

[6] Tatsuya Akutsu, Takeyuki Tamura, Daiji Fukagawa, and Atsuhiro Takasu. “Efficient exponential-time algorithms for edit distance between unordered trees”. In: *Journal of Discrete Algorithms* 25 (2014), pp. 79–93.

[7] Abdullah Al-Dujaili, François Merciol, and Sébastien Lefèvre. “GraphBPT: An efficient hierarchical data structure for image representation and probabilistic inference”. In: *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. 2015, pp. 301–312.

[8] Emanuel Aldea, Jamal Atif, and Isabelle Bloch. “Image classification using marginalized kernels for graphs”. In: *Graph-Based Representations in Pattern Recognition*. 2007, pp. 103–113.

[9] Argyros Argyridis and Demetre P Argialas. “A fuzzy spatial reasoner for multi-scale GEOBIA ontologies”. In: *Photogrammetric Engineering and Remote Sensing* 81.6 (2015), pp. 491–498.

[10] Peter M Atkinson and ARL Tatnall. “Introduction neural networks in remote sensing”. In: *International Journal of remote sensing* 18.4 (1997), pp. 699–709.

[11] Mariana Belgiu and Lucian Drăguţ. “Random forest in remote sensing: A review of applications and future directions”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 114 (2016), pp. 24–31.

[12] Jón Atli Benediktsson, Jocelyn Chanussot, and Mathieu Fauvel. “Multiple classifier systems in remote sensing: from basics to recent developments”. In: *International Workshop on Multiple Classifier Systems*. 2007, pp. 501–512.

[13] Ursula C Benz, Peter Hofmann, Gregor Willhauck, Iris Lingenfelder, and Markus Heynen. “Multi-resolution, object-oriented fuzzy analysis of remote sensing data for GIS-ready information”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 58.3 (2004), pp. 239–258.

[14] Philip Bille. “A survey on tree edit distance and related problems”. In: *Theoretical computer science* 337.1 (2005), pp. 217–239.

[15] José M Bioucas-Dias, Antonio Plaza, Gustavo Camps-Valls, Paul Scheunders, Nasser Nasrabadi, and Jocelyn Chanussot. “Hyperspectral remote sensing data analysis and future challenges”. In: *IEEE Geoscience and Remote Sensing Magazine* 1.2 (2013), pp. 6–36.

[16] Thomas Blaschke. “Object based image analysis for remote sensing”. In: *ISPRS journal of Photogrammetry and Remote Sensing* 65.1 (2010), pp. 2–16.

[17] Thomas Blaschke, Geoffrey J Hay, Maggi Kelly, Stefan Lang, Peter Hofmann, Elisabeth Addink, Raul Queiroz Feitosa, Freek van der Meer, Harald van der Werff, and Frieke van Coillie. “Geographic object-based image analysis–towards a new paradigm”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 87 (2014), pp. 180–191.

[18] Liefeng Bo, Xiaofeng Ren, and Dieter Fox. “Kernel descriptors for visual recognition”. In: *Advances in Neural Information Processing Systems*. 2010, pp. 244–252.

[19] Liefeng Bo and Cristian Sminchisescu. “Efficient match kernel between sets of features for visual recognition”. In: *Advances in Neural Information Processing Systems*. 2009, pp. 135–143.

[20] Karsten M Borgwardt and Hans-Peter Kriegel. “Shortest-path kernels on graphs”. In: *IEEE International Conference on Data Mining*. 2005, pp. 8–16.

[21] Karsten M Borgwardt, Cheng Soon Ong, Stefan Schönauer, SVN Vishwanathan, Alex J Smola, and Hans-Peter Kriegel. “Protein function prediction via graph kernels”. In: *Bioinformatics* 21.1 (2005), pp. 47–56.

[22] Petra Bosilj. “Image indexing and retrieval using component trees”. PhD thesis. Université de Bretagne Sud, 2016.

[23] Petra Bosilj, Erchan Aptoula, Sébastien Lefèvre, and Ewa Kijak. "Retrieval of remote sensing images with pattern spectra descriptors". In: *ISPRS International Journal of Geo-Information* 5.12 (2016).

[24] Gunnar Jakob Briem, Jón Atli Benediktsson, and Johannes R Sveinsson. "Multiple classifiers applied to multisource remote sensing data". In: *IEEE Transactions on Geoscience and Remote Sensing* 40.10 (2002), pp. 2291–2299.

[25] Lorenzo Bruzzone and Lorenzo Carlin. "A multilevel context-based system for classification of very high spatial resolution images". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.9 (2006), pp. 2587–2600.

[26] Serhat S Bucak, Rong Jin, and Anil K Jain. "Multiple kernel learning for visual object recognition: A review". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.7 (2014), pp. 1354–1369.

[27] Gustavo Camps-Valls and Lorenzo Bruzzone. *Kernel methods for remote sensing data analysis*. John Wiley & Sons, 2009.

[28] Gustavo Camps-Valls, Luis Gómez-Chova, Jordi Muñoz-Marí, José Luis Rojo-Álvarez, and Manel Martínez-Ramón. "Kernel-based framework for multitemporal and multisource remote sensing data classification and change detection". In: *IEEE Transactions on Geoscience and Remote Sensing* 46.6 (2008), pp. 1822–1835.

[29] Gustavo Camps-Valls, Devis Tuia, Luis Gómez-Chova, Sandra Jiménez, and Jesús Malo. "Remote sensing image processing". In: *Synthesis Lectures on Image, Video, and Multimedia Processing* 5.1 (2011), pp. 1–192.

[30] Gabriele Cavallaro, Mauro Dalla Mura, Jón Atli Benediktsson, and Antonio Plaza. "Remote sensing image classification using attribute filters defined over the Tree of Shapes". In: *IEEE Transactions on Geoscience and Remote Sensing* 54.7 (2016), pp. 3899–3911.

[31] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM: a library for support vector machines". In: *ACM Transactions on Intelligent Systems and Technology* 2.3 (2011), p. 27.

[32] Laetitia Chapel and Chloé Friguet. "Anomaly detection with score functions based on the reconstruction error of the kernel PCA". In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. 2014, pp. 227–241.

[33] Ken Chatfield, Victor S Lempitsky, Andrea Vedaldi, and Andrew Zisserman. "The devil is in the details: an evaluation of recent feature encoding methods". In: *British Machine Vision Conference*. 4. 2011, pp. 8–20.

[34] Lijun Chen, Wen Yang, Kan Xu, and Tao Xu. "Evaluation of local features for scene classification using VHR satellite images". In: *Joint Urban Remote Sensing Event*. 2011, pp. 385–388.

[35] Shizhi Chen and YingLi Tian. “Pyramid of spatial relatons for scene-level land use classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 53.4 (2015), pp. 1947–1957.

[36] Yunhao Chen, Wei Su, Jing Li, and Zhongping Sun. “Hierarchical object oriented classification using very high resolution imagery and LIDAR data over urban areas”. In: *Advances in Space Research* 43.7 (2009), pp. 1101–1110.

[37] Mingmin Chi, Antonio Plaza, Jón Atli Benediktsson, Zhongyi Sun, Jinsheng Shen, and Yangyong Zhu. “Big data for remote sensing: Challenges and opportunities”. In: *Proceedings of the IEEE* 104.11 (2016), pp. 2207–2219.

[38] Radha Chitta, Rong Jin, and Anil K Jain. “Efficient kernel clustering using random fourier features”. In: *IEEE 12th International Conference on Data Mining*. 2012, pp. 161–170.

[39] Michael Collins and Nigel Duffy. “Convolution kernels for natural language”. In: *Advances in Neural Information Processing Systems*. 2001, pp. 625–632.

[40] Russell G Congalton. “A review of assessing the accuracy of classifications of remotely sensed data”. In: *Remote Sensing of Environment* 37.1 (1991), pp. 35–46.

[41] Gianni Costa, Riccardo Ortale, and Ettore Ritacco. “X-class: Associative classification of XML documents by structure”. In: *ACM Transactions on Information Systems* 31.1 (2013), pp. 1 –40.

[42] Thomas Cover and Peter Hart. “Nearest neighbor pattern classification”. In: *IEEE Transactions on Information Theory* 13.1 (1967), pp. 21–27.

[43] Yanwei Cui, Laetitia Chapel, and Sébastien Lefèvre. “A subpath kernel for learning hierarchical image representations”. In: *International Workshop on Graph-Based Representations in Pattern Recognition*. 2015, pp. 34–43. DOI: 10.1007/978-3-319-18224-7_4.

[44] Yanwei Cui, Laetitia Chapel, and Sébastien Lefèvre. “Combining multiscale features for classification of hyperspectral images: a sequence based kernel approach”. In: *International Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*. 2016. URL: http://arxiv.org/abs/1606.04985.

[45] Yanwei Cui, Laetitia Chapel, and Sébastien Lefèvre. “Scalable bag of subpaths kernel for learning on hierarchical image representations and multi-source remote sensing data classification”. In: *Remote Sensing, Special Issue Advances in Object-Based Image Analysis—Linking with Computer Vision and Machine Learning* 9.3 (2017). DOI: 10.3390/rs9030196.

[46] Yanwei Cui, Sébastien Lefèvre, Laetitia Chapel, and Anne Puissant. “Combining multiple resolutions into hierarchical representations for kernel-based image classification”. In: *International Conference on Geographic Object-Based Image Analysis*. University of Twente, Enschede, The Netherlands, 2016. DOI: 10.3990/2.372.

[47] Marco Cuturi. “Fast global alignment kernels”. In: *Proceedings of the 28th International Conference on Machine Learning*. 2011, pp. 929–936.

[48] Mauro Dalla Mura, Jón Atli Benediktsson, Björn Waske, and Lorenzo Bruzzone. “Extended profiles with morphological attribute filters for the analysis of hyperspectral data”. In: *International Journal of Remote Sensing* 31.22 (2010), pp. 5975–5991.

[49] Mauro Dalla Mura, Jón Atli Benediktsson, Björn Waske, and Lorenzo Bruzzone. “Morphological attribute profiles for the analysis of very high resolution images”. In: *IEEE Transactions on Geoscience and Remote Sensing* 48.10 (2010), pp. 3747–3762.

[50] Mauro Dalla Mura, Saurabh Prasad, Fabio Pacifici, Paulo Gamba, Jocelyn Chanussot, and Jón Atli Benediktsson. “Challenges and opportunities of multimodality and data fusion in remote sensing”. In: *Proceedings of the IEEE* 103.9 (2015), pp. 1585–1601.

[51] Michele Dalponte, Lorenzo Bruzzone, and Damiano Gianelle. “Fusion of hyperspectral and LIDAR remote sensing data for classification of complex forest areas”. In: *IEEE Transactions on Geoscience and Remote Sensing* 46.5 (2008), pp. 1416–1427.

[52] Bharath Bhushan Damodaran, Joachim Höhle, and Sébastien Lefèvre. “Attribute profiles on derived features for urban land cover classification”. In: *Photogrammetric Engineering and Remote Sensing* 83.3 (2017), pp. 183–193.

[53] Bharath Bhushan Damodaran, Rama Rao Nidamanuri, and Yuliya Tarabalka. “Dynamic ensemble selection approach for hyperspectral image classification with joint spectral and spatial information”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8.6 (2015), pp. 2405–2417.

[54] Jonathan Delhumeau, Philippe-Henri Gosselin, Hervé Jégou, and Patrick Pérez. “Revisiting the VLAD image representation”. In: *ACM international conference on Multimedia*. 2013, pp. 653–656.

[55] Jefersson Alex Dos Santos, Philippe-Henri Gosselin, Sylvie Philipp-Foliguet, Ricardo da S Torres, and Alexandre Xavier Falao. “Multiscale classification of remote sensing images”. In: *IEEE Transactions on Geoscience and Remote Sensing* 50.10 (2012), pp. 3764–3775.

[56] Lucian Drăguţ, Dirk Tiede, and Shaun R Levick. “ESP: a tool to estimate scale parameter for multiresolution image segmentation of remotely sensed data”. In: *International Journal of Geographical Information Science* 24.6 (2010), pp. 859–871.

[57] François-Xavier Dupé and Luc Brun. “Tree covering within a graph kernel framework for shape classification”. In: *Image Analysis and Processing*. Springer, 2009, pp. 278–287.

[58] C Eisank, L Drăguţ, J Götz, and T Blaschke. “Developing a semantic model of glacial landforms for object-based terrain classification—the example of glacial cirques”. In: *GEOBIA* (2010), pp. 1682–1777.

[59] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. "LIBLINEAR: A library for large linear classification". In: *Journal of Machine Learning Research* 9.Aug (2008), pp. 1871–1874.

[60] Mathieu Fauvel, Jocelyn Chanussot, and Jón Atli Benediktsson. "A combined support vector machines classification based on decision fusion". In: *IEEE International Geoscience and Remote Sensing Symposium*. 2006, pp. 2494–2497.

[61] Mathieu Fauvel, Jocelyn Chanussot, and Jón Atli Benediktsson. "A spatial-spectral kernel-based approach for the classification of remote-sensing images". In: *Pattern Recognition* 45.1 (2012), pp. 381–392.

[62] Mathieu Fauvel, Yuliya Tarabalka, Jón Atli Benediktsson, Jocelyn Chanussot, and James C Tilton. "Advances in spectral-spatial classification of hyperspectral images". In: *Proceedings of the IEEE* 101.3 (2013), pp. 652–675.

[63] Aasa Feragen, Niklas Kasenburg, Jens Petersen, Marleen de Bruijne, and Karsten Borgwardt. "Scalable kernels for graphs with continuous attributes". In: *Advances in Neural Information Processing Systems*. 2013, pp. 216–224.

[64] Maurizio Filippone, Francesco Camastra, Francesco Masulli, and Stefano Rovetta. "A survey of kernel and spectral methods for clustering". In: *Pattern recognition* 41.1 (2008), pp. 176–190.

[65] Matthew Fisher, Manolis Savva, and Pat Hanrahan. "Characterizing structural relationships in scenes using graph kernels". In: *ACM Transactions on Graphics*. Vol. 30. 4. 2011, pp. 34–45.

[66] Germain Forestier, Anne Puissant, Cédric Wemmert, and Pierre Gançarski. "Knowledge-based region labeling for remote sensing image interpretation". In: *Computers, Environment and Urban Systems* 36.5 (2012), pp. 470–480.

[67] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Springer series in statistics Springer, Berlin, 2001.

[68] Holger Fröhlich, Jörg K Wegner, Florian Sieker, and Andreas Zell. "Optimal assignment kernels for attributed molecular graphs". In: *International Conference on Machine Learning*. 2005, pp. 225–232.

[69] Xinbo Gao, Bing Xiao, Dacheng Tao, and Xuelong Li. "A survey of graph edit distance". In: *Pattern Analysis and applications* 13.1 (2010), pp. 113–129.

[70] Valeria Garro and Andrea Giachetti. "Scale space graph representation and kernel matching for non rigid and textured 3D shape retrieval". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.6 (2016), pp. 1258–1271.

[71] Thomas Gärtner. "A survey of kernels for structured data". In: *ACM Special Interest Group on Knowledge Discovery in Data Explorations Newsletter* 5.1 (2003), pp. 49–58.

[72] Thomas Gärtner, Peter Flach, and Stefan Wrobel. "On graph kernels: Hardness results and efficient alternatives". In: *Learning Theory and Kernel Machines*. Springer, 2003, pp. 129–143.

[73] Benoit Gaüzère, Pierre-Anthony Grenier, Luc Brun, and Didier Villemin. "Treelet kernel incorporating cyclic, stereo and inter pattern information in chemoinformatics". In: *Pattern Recognition* 48.2 (2015), pp. 356–367.

[74] Thierry Géraud, Edwin Carlinet, Sébastien Crozet, and Laurent Najman. "A quasi-linear algorithm to compute the tree of shapes of nD images". In: *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. 2013, pp. 98–110.

[75] Pedram Ghamisi, Jón Atli Benediktsson, and Johannes R Sveinsson. "Automatic spectral-spatial classification framework based on attribute profiles and supervised feature extraction". In: *IEEE Transactions on Geoscience and Remote Sensing* 52.9 (2014), pp. 5771–5782.

[76] Pedram Ghamisi, Mauro Dalla Mura, and Jón Atli Benediktsson. "A survey on spectral-spatial classification techniques based on attribute profiles". In: *IEEE Transactions on Geoscience and Remote Sensing* 53.5 (2015), pp. 2335–2353.

[77] Demir Gokalp and Selim Aksoy. "Scene classification using bag-of-regions representations". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2007, pp. 1–8.

[78] Luis Gomez-Chova, Devis Tuia, Gabriele Moser, and Gustau Camps-Valls. "Multimodal classification of remote sensing images: a review and future directions". In: *Proceedings of the IEEE* 103.9 (2015), pp. 1560–1584.

[79] Mehmet Gönen and Ethem Alpaydın. "Multiple kernel learning algorithms". In: *Journal of Machine Learning Research* 12 (2011), pp. 2211–2268.

[80] Philippe-Henri Gosselin, Matthieu Cord, and Sylvie Philipp-Foliguet. "Kernels on bags for multi-object database retrieval". In: *ACM International Conference on Image and Video Retrieval*. 2007, pp. 226–231.

[81] Pierre-Anthony Grenier, Luc Brun, and Didier Villemin. "Chemoinformatics and stereoisomerism: A stereo graph kernel together with three new extensions". In: *Pattern Recognition Letters* (2016), pp. 222 –230.

[82] Chunhui Gu, Joseph J Lim, Pablo Arbeláez, and Jitendra Malik. "Recognition using regions". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 1030–1037.

[83] Lionel Gueguen and Georgios K Ouzounis. "Hierarchical data representation structures for interactive image information mining". In: *International Journal of Image and Data Fusion* 3.3 (2012), pp. 221–241.

[84] James Hafner, Harpreet S. Sawhney, William Equitz, Myron Flickner, and Wayne Niblack. "Efficient color histogram indexing for quadratic form distance functions". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17.7 (1995), pp. 729–736.

[85] Issei Hamada, Takaharu Shimada, Daiki Nakata, Kouichi Hirata, and Tetsuji Kuboyama. "Agreement subtree mapping kernel for phylogenetic trees". In: *New Frontiers in Artificial Intelligence*. 2014, pp. 321–336.

[86] Robert M Haralick, Karthikeyan Shanmugam, and Itshak Dinstein. "Textural features for image classification". In: *IEEE Transactions on Systems, Man, and Cybernetics* 3.6 (1973), pp. 610–621.

[87] Zaïd Harchaoui and Francis Bach. "Image classification with segmentation graph kernels". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2007, pp. 1–8.

[88] David Haussler. *Convolution kernels on discrete structures*. Tech. rep. Department of Computer Science, University of California at Santa Cruz, 1999.

[89] Dong-Chen He and Li Wang. "Texture unit, texture spectrum, and texture analysis". In: *IEEE Transactions on Geoscience and Remote Sensing* 28.4 (1990), pp. 509–512.

[90] Ihsen Hedhli, Gabriele Moser, Sebastiano Serpico, and Josiane Zerubia. "Multi-resolution classification of urban areas using hierarchical symmetric Markov mesh models". In: *Joint Urban Remote Sensing Event*. 2017.

[91] Ihsen Hedhli, Gabriele Moser, Josiane Zerubia, and Sebastiano B Serpico. "Fusion of multitemporal and multiresolution remote sensing data and application to natural disasters". In: *IEEE International Geoscience and Remote Sensing Symposium*. 2014, pp. 207–210.

[92] Kouichi Hirata, Tetsuji Kuboyama, and Takuya Yoshino. "Mapping kernels between rooted labeled trees beyond ordered trees". In: *New Frontiers in Artificial Intelligence*. Vol. 9067. 2015, pp. 317–330.

[93] John E Hopcroft, Jeffrey David Ullman, and Alfred Vaino Aho. *Data structures and algorithms*. Addison-Wesley Boston, MA, USA: 1983.

[94] Fan Hu, Gui-Song Xia, Jingwen Hu, and Liangpei Zhang. "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery". In: *Remote Sensing* 7.11 (2015), pp. 14680–14707.

[95] Jingwen Hu, Gui-Song Xia, Fan Hu, and Liangpei Zhang. "A comparative study of sampling analysis in the scene classification of optical high-spatial resolution remote sensing imagery". In: *Remote Sensing* 7.11 (2015), pp. 14988–15013.

[96] Lian-Zhi Huo, Ping Tang, Zheng Zhang, and Devis Tuia. "Semisupervised classification of remote sensing images with hierarchical spatial similarity". In: *IEEE Geoscience and Remote Sensing Letters* 12.1 (2015), pp. 150–154.

[97] Mayank Juneja, Andrea Vedaldi, CV Jawahar, and Andrew Zisserman. "Blocks that shout: Distinctive parts for scene classification". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 923–930.

[98] Hisashi Kashima. "Machine Learning Approaches for Structured Data". PhD thesis. Japan: Graduate School of Informatics, Kyoto University, 2007.

[99] Hisashi Kashima and Teruo Koyanagi. "Kernels for Semi-Structured Data". In: *International Conference on Machine Learning*. 2002, pp. 291–298.

[100] Hisashi Kashima, Koji Tsuda, and Akihiro Inokuchi. "Marginalized kernels between labeled graphs". In: *International Conference on Machine Learning*. Vol. 3. 2003, pp. 321–328.

[101] Daisuke Kimura and Hisashi Kashima. "Fast computation of subpath kernel for trees". In: *International Conference on Machine Learning*. 2012, pp. 393–400.

[102] Daisuke Kimura, Tetsuji Kuboyama, Tetsuo Shibuya, and Hisashi Kashima. "A subpath kernel for rooted unordered trees". In: *Advances in Knowledge Discovery and Data Mining*. Springer, 2011, pp. 62–74.

[103] Piotr Koniusz, Fei Yan, Philippe-Henri Gosselin, and Krystian Mikolajczyk. "Higher-order occurrence pooling for bags-of-words: Visual concept detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.2 (2017), pp. 313–326.

[104] Nils Kriege, Marion Neumann, Kristian Kersting, and Petra Mutzel. "Explicit versus implicit graph feature maps: A computational phase transition for walk kernels". In: *IEEE International Conference on Data Mining*. 2014, pp. 881–886.

[105] Nils M Kriege, Pierre-Louis Giscard, and Richard Wilson. "On valid optimal assignment kernels and applications to graph classification". In: *Advances in Neural Information Processing Systems*. 2016, pp. 1615–1623.

[106] Nils M Kriege, Marion Neumann, Christopher Morris, Kristian Kersting, and Petra Mutzel. "A unifying view of explicit and implicit feature maps for structured data: systematic studies of graph kernels". In: *arXiv preprint arXiv:1703.00676* (2017).

[107] Camille Kurtz, Nicolas Passat, Pierre Gancarski, and Anne Puissant. "Extraction of complex patterns from multiresolution remote sensing images: A hierarchical top-down methodology". In: *Pattern Recognition* 45.2 (2012), pp. 685–706.

[108] Heesung Kwon and Nasser M Nasrabadi. "Kernel RX-algorithm: A nonlinear anomaly detector for hyperspectral imagery". In: *IEEE Transactions on Geoscience and Remote Sensing* 43.2 (2005), pp. 388–397.

[109] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories". In: *IEEE Conference on Computer vision and Pattern Recognition*. Vol. 2. 2006, pp. 2169–2178.

[110] Justine Lebrun, Philippe-Henri Gosselin, and Sylvie Philipp-Foliguet. "Inexact graph matching based on kernels for object retrieval in image databases". In: *Image and Vision Computing* 29.11 (2011), pp. 716–729.

[111] Justine Lebrun, Sylvie Philipp-Foliguet, and Philippe-Henri Gosselin. “Image retrieval with graph kernel on regions”. In: *19th International Conference on Pattern Recognition.* 2008, pp. 1–4.

[112] Chen-Yu Lee, Anurag Bhardwaj, Wei Di, Vignesh Jagadeesh, and Robinson Piramuthu. “Region-based discriminative feature pooling for scene text recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 4050–4057.

[113] Sébastien Lefèvre, Laëtitia Chapel, and François Merciol. “Hyperspectral image classification from multiscale description with constrained connectivity and metric learning”. In: *6th International Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*. 2014.

[114] Sébastien Lefèvre, Devis Tuia, Jan Dirk Wegner, Timothée Produit, and Ahmed Samy Nassar. “Toward Seamless Multiview Scene Analysis From Satellite to Street Level”. In: *Proceedings of the IEEE* (2017).

[115] Christina Leslie, Eleazar Eskin, Jason Weston, and William Stafford Noble. “Mismatch string kernels for SVM protein classification”. In: *Advances in Neural Information Processing Systems*. 2002, pp. 1441–1448.

[116] Fuxin Li, Catalin Ionescu, and Cristian Sminchisescu. “Random Fourier approximations for skewed multiplicative histogram kernels”. In: *Joint Pattern Recognition Symposium*. 2010, pp. 262–271.

[117] Fuxin Li, Guy Lebanon, and Cristian Sminchisescu. “Chebyshev approximations to the histogram χ 2 kernel”. In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 2424–2431.

[118] Songnian Li, Suzana Dragicevic, Francesc Antón Castro, Monika Sester, Stephan Winter, Arzu Coltekin, Christopher Pettit, Bin Jiang, James Haworth, and Alfred Stein. “Geospatial big data handling theory and methods: A review and research challenges”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 115 (2016), pp. 119–133.

[119] Wenzhi Liao, Xin Huang, Frieke Van Coillie, Sidharta Gautama, Aleksandra Pižurica, Wilfried Philips, Hui Liu, Tingting Zhu, Michal Shimoni, and Gabriele Moser. “Processing of multiresolution thermal hyperspectral and digital color data: Outcome of the 2014 IEEE GRSS data fusion contest”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8.6 (2015), pp. 2984–2996.

[120] Tianzhu Liu, Yanfeng Gu, Xiuping Jia, Jón Atli Benediktsson, and Jocelyn Chanussot. “Class-Specific Sparse Multiple Kernel Learning for Spectral–Spatial Hyperspectral Image Classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 54.12 (2016), pp. 7351–7365.

[121] Yu Liu, Qinghua Guo, and Maggi Kelly. “A framework of region-based spatial relations for non-overlapping features and its application in object based image analysis”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 63.4 (2008), pp. 461–475.

[122] David Lopez-Paz, Suvrit Sra, Alex Smola, Zoubin Ghahramani, and Bernhard Schölkopf. “Randomized nonlinear component analysis”. In: *International Conference on Machine Learning*. 2014, pp. 1359–1367.

[123] David G Lowe. “Distinctive image features from scale-invariant keypoints”. In: *International journal of Computer Vision* 60.2 (2004), pp. 91–110.

[124] Zhiwu Lu and Horace HS Ip. “Image categorization with spatial mismatch kernels”. In: *IEEE Conference on Computer Vision and Pattern Recognition.* 2009, pp. 397–404.

[125] Zhiyun Lu, Avner May, Kuan Liu, Alireza Bagheri Garakani, Dong Guo, Aurélien Bellet, Linxi Fan, Michael Collins, Brian Kingsbury, and Michael Picheny. “How to scale up kernel methods to be as good as deep neural nets”. In: *arXiv preprint arXiv:1411.4000* (2014).

[126] Yan Ma, Haiping Wu, Lizhe Wang, Bormin Huang, Rajiv Ranjan, Albert Zomaya, and Wei Jie. “Remote sensing big data computing: challenges and opportunities”. In: *Future Generation Computer Systems* 51 (2015), pp. 47–60.

[127] Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez. “Convolutional neural networks for large-Scale remote-sensing image classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 55.2 (2017), pp. 645–657.

[128] Pierre Mahé and Jean-Philippe Vert. “Graph kernels based on tree patterns for molecules”. In: *Machine learning* 75.1 (2009), pp. 3–35.

[129] Javier Marcello, A Medina, and Francisco Eugenio. “Evaluation of spatial and spectral effectiveness of pixel-level fusion techniques”. In: *IEEE Geoscience and remote sensing letters* 10.3 (2013), pp. 432–436.

[130] Pier Giorgio Marchetti, Pierre Soille, and Lorenzo Bruzzone. “A special issue on big data from space for geoscience and remote sensing [from the guest editors]”. In: *IEEE Geoscience and Remote Sensing Magazine* 4.3 (2016), pp. 7–9.

[131] Giovanni Da San Martino, Nicolò Navarin, and Alessandro Sperduti. “A tree-based kernel for graphs with continuous attributes”. In: *arXiv preprint arXiv:1509.01116* (2015).

[132] G Meinel and M Neubert. “A comparison of segmentation programs for high resolution remote sensing data”. In: *International Archives of Photogrammetry and Remote Sensing* 35 (2004), pp. 1097–1105.

[133] Farid Melgani and Lorenzo Bruzzone. “Classification of hyperspectral remote sensing images with support vector machines”. In: *IEEE Transactions on Geoscience and Remote Sensing* 42.8 (2004), pp. 1778–1790.

[134] Volodymyr Mnih. “Machine learning for aerial image labeling”. PhD thesis. University of Toronto, 2013.

[135] Arnaud Poncet Montanges, Gabriele Moser, Hannes Taubenböck, Michael Wurm, and Devis Tuia. “Classification of urban structural types with multisource data and structured models”. In: *2015 Joint Urban Remote Sensing Event*. 2015, pp. 1–4.

[136] Christopher Morris, Nils M Kriege, Kristian Kersting, and Petra Mutzel. “Faster kernels for graphs with continuous attributes via hashing”. In: *arXiv preprint arXiv:1610.00064* (2016).

[137] Alessandro Moschitti. “Efficient convolution kernels for dependency and constituent syntactic trees”. In: *European Conference on Machine Learning*. 2006, pp. 318–329.

[138] Gabriele Moser, Sebastiano B Serpico, and Jón Atli Benediktsson. “Land-cover mapping by Markov modeling of spatial–contextual information in very-high-resolution remote sensing images”. In: *Proceedings of the IEEE* 101.3 (2013), pp. 631–651.

[139] Giorgos Mountrakis, Jungho Im, and Caesar Ogole. “Support vector machines in remote sensing: A review”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 66.3 (2011), pp. 247–259.

[140] Naila Murray, Hervé Jégou, Florent Perronnin, and Andrew Zisserman. “Interferences in match kernels”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2016).

[141] Naila Murray and Florent Perronnin. “Generalized max pooling”. In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 2473–2480.

[142] Laurent Najman and Michel Couprie. “Building the component tree in quasi-linear time”. In: *IEEE Transactions on Image Processing* 15.11 (2006), pp. 3531–3539.

[143] Romain Negrel, David Picard, and Philippe-Henri Gosselin. “Evaluation of second-order visual features for land-use classification”. In: *International Workshop on Content-Based Multimedia Indexing*. 2014, pp. 1–5.

[144] Michel Neuhaus and Horst Bunke. “Edit distance-based kernel functions for structural pattern classification”. In: *Pattern Recognition* 39.10 (2006), pp. 1852–1863.

[145] Marion Neumann, Roman Garnett, Christian Bauckhage, and Kristian Kersting. “Propagation kernels: efficient graph kernels from propagated information”. In: *Machine Learning* 102.2 (2016), pp. 209–245.

[146] Sebastian Nowozin and Christoph H. Lampert. “Structured learning and prediction in computer vision”. In: *Foundations and Trends in Computer Graphics and Vision* 6.3–4 (2011), pp. 185–365.

[147] Aude Oliva and Antonio Torralba. “Modeling the shape of the scene: A holistic representation of the spatial envelope”. In: *International Journal of Computer Vision* 42.3 (2001), pp. 145–175.

[148] Georgios K Ouzounis and Pierre Soille. "Pattern spectra from partition pyramids and hierarchies". In: *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. 2011, pp. 108–119.

[149] Fabio Pacifici, Nathan Longbotham, and William J Emery. "The importance of physical quantities for the analysis of multitemporal and multiangular optical very high spatial resolution images". In: *IEEE Transactions on Geoscience and Remote Sensing* 52.10 (2014), pp. 6241–6256.

[150] Mahesh Pal. "Random forest classifier for remote sensing classification". In: *International Journal of Remote Sensing* 26.1 (2005), pp. 217–222.

[151] Edoardo Pasolli, Farid Melgani, Devis Tuia, Fabio Pacifici, and William J Emery. "SVM active learning approach for image classification using spatial information". In: *IEEE Transactions on Geoscience and Remote Sensing* 52.4 (2014), pp. 2217–2233.

[152] Mattia Pedergnana, Prashanth Reddy Marpu, Mauro Dalla Mura, Jón Atli Benediktsson, and Lorenzo Bruzzone. "A novel technique for optimal feature selection in attribute profiles based on genetic algorithms". In: *IEEE Transactions on Geoscience and Remote Sensing* 51.6 (2013), pp. 3514–3528.

[153] Otávio AB Penatti, Eduardo Valle, and Ricardo da S Torres. "Comparative study of global color and texture descriptors for web image retrieval". In: *Journal of Visual Communication and Image Representation* 23.2 (2012), pp. 359–380.

[154] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. "Improving the fisher kernel for large-scale image classification". In: *In Proceedings of the European Conference on Computer Vision*. 2010, pp. 143–156.

[155] Florent Perronnin, Jorge Sénchez, and Yan Liu Xerox. "Large-scale image categorization with explicit data embedding". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2010, pp. 2297–2304.

[156] Nils Plath, Marc Toussaint, and Shinichi Nakajima. "Multi-class image segmentation using conditional random fields and global classification". In: *International Conference on Machine Learning*. 2009, pp. 817–824.

[157] Antonio Plaza, Jón Atli Benediktsson, Joseph W Boardman, Jason Brazile, Lorenzo Bruzzone, Gustavo Camps-Valls, Jocelyn Chanussot, Mathieu Fauvel, Paolo Gamba, Anthony Gualtieri, et al. "Recent advances in techniques for hyperspectral image processing". In: *Remote sensing of environment* 113 (2009), S110–S122.

[158] Anne Puissant, Nicolas Durand, David Sheeren, Christiane Weber, and Pierre Gancarski. "Urban ontology for semantic interpretation of multi-source images". In: *2nd Workshop Ontologies for Urban Development: Conceptual Models for Practitioners*. 2007.

[159] Cheng Qiao, Jinfei Wang, Jiali Shang, and Bahram Daneshfar. "Spatial relationship-assisted classification from high-resolution remote sensing imagery". In: *International Journal of Digital Earth* 8.9 (2015), pp. 710–726.

[160] Ali Rahimi and Benjamin Recht. "Random features for large-scale kernel machines". In: *Advances in Neural Information Processing Systems*. 2007, pp. 1177–1184.

[161] Ali Rahimi and Benjamin Recht. "Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning". In: *Advances in Neural Information Processing Systems*. 2009, pp. 1313–1320.

[162] Jordan Reynolds and Kevin Murphy. "Figure-ground segmentation using a hierarchical conditional random field". In: *IEEE Canadian Conference on Computer and Robot Vision*. IEEE. 2007, pp. 175–182.

[163] Eric Sven Ristad and Peter N Yianilos. "Learning string-edit distance". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.5 (1998), pp. 522–532.

[164] Philippe Salembier and Luis Garrido. "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval". In: *IEEE Transactions on Image Processing* 9.4 (2000), pp. 561–576.

[165] Konrad Schindler. "An overview and comparison of smooth labeling methods for land-cover classification". In: *IEEE Transactions on Geoscience and Remote Sensing* 50.11 (2012), pp. 4534–4545.

[166] John Shawe-Taylor and Nello Cristianini. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[167] Nino Shervashidze, SVN Vishwanathan, Tobias Petri, Kurt Mehlhorn, and Karsten M Borgwardt. "Efficient graphlet kernels for large graph comparison". In: *International Conference on Artificial Intelligence and Statistics*. 2009, pp. 488–495.

[168] Kilho Shin and Tetsuji Kuboyama. "A comprehensive study of tree kernels". In: *New Frontiers in Artificial Intelligence*. 2014, pp. 337–351.

[169] Kilho Shin and Tetsuji Kuboyama. "A generalization of Haussler's convolution kernel: mapping kernel". In: *International Conference on Machine Learning*. 2008, pp. 944–951.

[170] Rajesh BV Shruthi, Norman Kerle, and Victor Jetten. "Object-based gully feature extraction using high spatial resolution imagery". In: *Geomorphology* 134.3 (2011), pp. 260–268.

[171] Aman Sinha and John C Duchi. "Learning kernels with random features". In: *Advances In Neural Information Processing Systems*. 2016, pp. 1298–1306.

[172] Leen-Kiat Soh and Costas Tsatsoulis. "Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices". In: *IEEE Transactions on geoscience and remote sensing* 37.2 (1999), pp. 780–795.

[173] Pierre Soille. "Constrained connectivity for hierarchical image partitioning and simplification". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.7 (2008), pp. 1132–1145.

[174] Anne H Schistad Solberg, Torfinn Taxt, and Anil K Jain. "A Markov random field model for classification of multisource satellite imagery". In: *IEEE Transactions on Geoscience and Remote Sensing* 34.1 (1996), pp. 100–113.

[175] Shashank Srivastava, Dirk Hovy, and Eduard H. Hovy. "A walk-based semantically enriched tree kernel over distributed word representations". In: *Conference on Empirical Methods in Natural Language Processing*. 2013, pp. 1411–1416.

[176] Geir Storvik, R Fjortoft, and Anne H Schistad Solberg. "A Bayesian approach to classification of multiresolution remote sensing data". In: *IEEE Transactions on Geoscience and Remote Sensing* 43.3 (2005), pp. 539–547.

[177] Frédéric Suard, Vincent Guigue, Alain Rakotomamonjy, and Abdelaziz Benshrair. "Pedestrian detection using stereo-vision and graph kernels". In: *Intelligent Vehicles Symposium*. 2005, pp. 267–272.

[178] Frédéric Suard, Alain Rakotomamonjy, and Abdelaziz Benrshrair. "Kernel on Bag of Paths For Measuring Similarity of Shapes". In: *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*. 2007, pp. 355–360.

[179] Frédéric Suard, Alain Rakotomamonjy, and Abdelaziz Bensrhair. "Object categorization using kernels combining graphs and histograms of gradients". In: *Image Analysis and Recognition*. Springer, 2006, pp. 23–34.

[180] Dougal J Sutherland and Jeff Schneider. "On the error of random Fourier features". In: *arXiv preprint arXiv:1506.02785* (2015).

[181] MA Tahir, Josef Kittler, K Mikolajczyk, F Yan, Koen EA van de Sande, and Theo Gevers. "Visual category recognition using spectral regression and kernel discriminant analysis". In: *IEEE International Conference on Computer Vision Workshops*. 2009, pp. 178–185.

[182] Kuo-Chung Tai. "The tree-to-tree correction problem". In: *Journal of the ACM* 26.3 (1979), pp. 422–433.

[183] Yuliya Tarabalka, Jón Atli Benediktsson, and Jocelyn Chanussot. "Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques". In: *IEEE Transactions on Geoscience and Remote Sensing* 47.8 (2009), pp. 2973–2987.

[184] Yuliya Tarabalka, Mathieu Fauvel, Jocelyn Chanussot, and Jón Atli Benediktsson. "SVM-and MRF-based method for accurate classification of hyperspectral images". In: *IEEE Geoscience and Remote Sensing Letters* 7.4 (2010), pp. 736–740.

[185] James C. Tilton. “Image segmentation by region growing and spectral clustering with a natural convergence criterion”. In: *IEEE International Geoscience and Remote Sensing Symposium*. 1998, pp. 1766–1768.

[186] Giorgos Tolias, Yannis Avrithis, and Hervé Jégou. “To aggregate or not to aggregate: Selective match kernels for image search”. In: *International Conference on Computer Vision*. 2013, pp. 1401–1408.

[187] Andrea Torsello, Dzena Hidovic-Rowe, and Marcello Pelillo. “Polynomial-time metrics for attributed trees”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.7 (2005), pp. 1087–1099.

[188] Devis Tuia, Gustavo Camps-Valls, Giona Matasci, and Mikhail Kanevski. “Learning relevant image features with multiple-kernel classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 48.10 (2010), pp. 3780–3791.

[189] Devis Tuia, Rémi Flamary, and Nicolas Courty. “Multiclass feature learning for hyperspectral image classification: Sparse and hierarchical solutions”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 105 (2015), pp. 272–285.

[190] Devis Tuia, Gabriele Moser, Bertrand Le Saux, Benjamin Bechtel, and Linda See. “2017 IEEE GRSS Data Fusion Contest: Open Data for Global Multimodal Land Use Classification [Technical Committees]”. In: *IEEE Geoscience and Remote Sensing Magazine* 5.1 (2017), pp. 70–73.

[191] Devis Tuia, Jordi Muñoz-Marí, Mikhail Kanevski, and Gustavo Camps-Valls. “Structured output SVM for remote sensing image classification”. In: *Journal of signal processing systems* 65.3 (2011), pp. 301–310.

[192] Devis Tuia, Claudio Persello, and Lorenzo Bruzzone. “Domain adaptation for the classification of remote sensing data: an overview of recent advances”. In: *IEEE Geoscience and Remote Sensing Magazine* 4.2 (2016), pp. 41–57.

[193] Devis Tuia, Frédéric Ratle, Fabio Pacifici, Mikhail F Kanevski, and William J Emery. “Active learning methods for remote sensing image classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 47.7 (2009), pp. 2218–2232.

[194] Devis Tuia, Frédéric Ratle, Alexei Pozdnoukhov, and Gustavo Camps-Valls. “Multisource composite kernels for urban-image classification”. In: *IEEE Geoscience and Remote Sensing Letters* 7.1 (2010), pp. 88–92.

[195] Devis Tuia, Michele Volpi, Mauro Dalla Mura, Alain Rakotomamonjy, and Rémi Flamary. “Automatic feature learning for spatio-spectral image classification with sparse SVM”. In: *IEEE Transactions on Geoscience and Remote Sensing* 52.10 (2014), pp. 6062–6074.

[196] Devis Tuia, Michele Volpi, and Gabriele Moser. "Getting pixels and regions to agree with conditional random fields". In: *IEEE International Geoscience and Remote Sensing Symposium*. 2016, pp. 3290–3293.

[197] Silvia Valero, Philippe Salembier, and Jocelyn Chanussot. "Hyperspectral image representation and processing with binary partition trees". In: *IEEE Transactions on Image Processing* 22.4 (2013), pp. 1430–1443.

[198] Andrea Vedaldi and Andrew Zisserman. "Efficient additive kernels via explicit feature maps". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.3 (2012), pp. 480–492.

[199] Jean-Philippe Vert. "The optimal assignment kernel is not positive definite". In: *arXiv preprint arXiv:0801.4061* (2008).

[200] Jean-Philippe Vert, Tomoko Matsui, Shin'ichi Satoh, and Yuji Uchiyama. "High-level feature extraction using SVM with walk-based graph kernel". In: *International Conference on Acoustics, Speech and Signal Processing*. 2009, pp. 1121–1124.

[201] S.V. N. Vishwanathan, Nicol N. Schraudolph, Risi Kondor, and Karsten M. Borgwardt. "Graph kernels". In: *Journal of Machine Learning Research* 11 (2010), pp. 1201–1242.

[202] SVN Vishwanathan and Alexander Johannes Smola. "Fast kernels for string and tree matching". In: *Kernel Methods in Computational Biology* (2004), pp. 113–130.

[203] Aurélie Voisin, Vladimir A Krylov, Gabriele Moser, Sebastiano B Serpico, and Josiane Zerubia. "Supervised classification of multisensor and multiresolution remote sensing images with a hierarchical copula-based approach". In: *IEEE Transactions on Geoscience and remote sensing* 52.6 (2014), pp. 3346–3358.

[204] Michele Volpi and Vittorio Ferrari. "Semantic segmentation of urban scenes by learning local class interactions". In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2015, pp. 1–9.

[205] Björn Waske and Jón Atli Benediktsson. "Fusion of support vector machines for classification of multisensor data". In: *IEEE Transactions on Geoscience and Remote Sensing* 45.12 (2007), pp. 3858–3866.

[206] Cédric Wemmert, Anne Puissant, Germain Forestier, and Pierre Gancarski. "Multiresolution remote sensing image clustering". In: *IEEE Geoscience and Remote Sensing Letters* 6.3 (2009), pp. 533–537.

[207] Jason Weston, Christina Leslie, Eugene Ie, Dengyong Zhou, Andre Elisseeff, and William Stafford Noble. "Semi-supervised protein classification using cluster kernels". In: *Bioinformatics* 21.15 (2005), pp. 3241–3247.

[208] Graeme G Wilkinson. "Results and implications of a study of fifteen years of satellite image classification experiments". In: *IEEE Transactions on Geoscience and Remote Sensing* 43.3 (2005), pp. 433–440.

[209] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang. "Linear spatial pyramid matching using sparse coding for image classification". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 1794–1801.

[210] Jiyan Yang, Vikas Sindhwani, Haim Avron, and Michael Mahoney. "Quasi-Monte Carlo feature maps for shift-invariant kernels". In: *International Conference on Machine Learning*. 2014, pp. 485–493.

[211] Michael Ying Yang and Wolfgang Förstner. "A hierarchical conditional random field model for labeling and classifying images of man-made scenes". In: *IEEE International Conference on Computer Vision Workshops*. 2011, pp. 196–203.

[212] Tianbao Yang, Yu-Feng Li, Mehrdad Mahdavi, Rong Jin, and Zhi-Hua Zhou. "Nyström method vs random fourier features: A theoretical and empirical comparison". In: *Advances in Neural Information Processing Systems*. 2012, pp. 476–484.

[213] Yi Yang and Shawn Newsam. "Bag-of-visual-words and spatial extensions for land-use classification". In: *International Conference on Advances in Geographic Information Systems*. 2010, pp. 270–279.

[214] Yi Yang and Shawn Newsam. "Spatial pyramid co-occurrence for image classification". In: *IEEE International Conference on Computer Vision*. 2011, pp. 1465–1472.

[215] Jian Yao, Sanja Fidler, and Raquel Urtasun. "Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 702–709.

[216] Huai Yu, Wen Yang, Gui-Song Xia, and Gang Liu. "A color-texture-structure descriptor for high-resolution satellite image classification". In: *Remote Sensing* 8.3 (2016).

[217] Jianguo Zhang, Marcin Marszałek, Svetlana Lazebnik, and Cordelia Schmid. "Local features and kernels for classification of texture and object categories: A comprehensive study". In: *International Journal of Computer Vision* 73.2 (2007), pp. 213–238.

[218] Jixian Zhang. "Multi-source remote sensing data fusion: status and trends". In: *International Journal of Image and Data Fusion* 1 (2010), pp. 5–24.

[219] Min Zhang, Wanxiang Che, GuoDong Zhou, Aiti Aw, Chew Lim Tan, Ting Liu, and Sheng Li. "Semantic role labeling using a grammar-driven convolution tree kernel". In: *IEEE Transactions on Audio, Speech, and Language Processing* 16.7 (2008), pp. 1315–1329.

[220] Yasen Zhang, Xian Sun, Hongqi Wang, and Kun Fu. "High-resolution remote-sensing image classification via an approximate earth mover's distance-based bag-of-features model". In: *IEEE Geoscience and Remote Sensing Letters* 10.5 (2013), pp. 1055–1059.

[221] Bei Zhao, Yanfei Zhong, and Liangpei Zhang. “A spectral-structural bag-of-features scene classifier for very high spatial resolution remote sensing imagery”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 116 (2016), pp. 73–85.